

特集 マシン語に挑戦する!

全リスト
一挙公開

✳️ TK-80 ミニ・アセンブラ & トレーサー

✳️ LKIT-16 逆アセンブラ

新連載 BASICの秘宝

楽しいゲーム TK-80BS MOO ゲーム & TV-01 数あてゲーム

アマチュア無線 8080 モールス符号発生 H68/TR モールス符号解説

1チップCPU 8086 & 8035 製作・電源・F/V/V/F



保存版

1978 6

定価—380yen

高性能、高品質、低価格——プロ仕様ターミナル。



パーソナルコンピューター

COSMO TERMINAL-D

六大特長

- 各種プリンター、フロッピーディスク、デジタルカセット等の周辺装置が完備しています。
 - 標準アクセサリ
 - ・コスモプリンター ￥295,000(有償)
 - ・TEAC MT-2 ￥95,000
 - ・ミニフロッピー(デュアル) ￥549,000
 - ・フロッピー(IBM) ￥169,000
 - キーボードはホール素子(無接点)
 - カラーディスプレイ カラーモニターに7色のディスプレイ もちろん家庭用のテレビにも接続可能です。
 - 開発用ツールとしてP-ROMライターを実装——内部拡張64Kまで(外部拡張は無限)
 - アセンブラーの他、BASIC、PLI、RPG、COBOL、FORTRAN等高級言語まで使用できます。
 - すぐれた拡張性とフレキシビリティ。
- 群を抜く汎用性——事務用・研究用・建設(土木)用、商店の在庫管理、振替の計算管理(保険点数など)、振替計算、アマチュアサイドのホビーなどワイド。
- MODEL-01/ASC11・ユーザー用RAM1K実装・モニターTVなし ￥299,000
 - MODEL-02/ASC11・ユーザー用RAM実装16K・モニターTVなし ￥399,000
 - MODEL-11/ASC11・ユーザー用RAM1K実装・モニターTV付(モノクローム) ￥349,000 ●MODEL-12/ASC11・ユーザー用RAM実装16K・モニターTV付(モノクローム) ￥449,000
 - カラーモニターTVは家庭用TVが使用できます。●価格はずべて完成品です。●キッパ販売はいたしません。●化粧ケース別売 ●COSMOS店での引渡し以外は送料￥15,000円となりますのであらかじめ送料加算の上 MODELNoと共に申し込みたい。●本製品の品扱いを希望する会社または業者の方はお近くのCOSMOS店にご相談下さい。

〈コスモターミナルD仕様〉

- CRT部
 - アルファ numeric メリットフルキーボード+CRTコントローラ(ASC11型)
 - P-ROMライター(MB8518インテル2706相当)実装
 - 機内式ホリスライダ全面使用
 - 双方向性 TV-RAM方式
 - ライトペンレススター内蔵
 - 7X9DOT MATRIX
 - 64X16+32X16切替式(キーボードより)
 - 文字カラー7色・ヘルトカラー7色・白黒反転・ラインレス機能
 - スクローリング・ペーパー転送切替式(キーボードより)
 - 75~9600ヘルツ切替式(キーボードより)
 - テレビアンプインターフェイス(20m ACL: ASR33コンパチブル)内蔵
 - ビデオカセットインターフェイス(カンサスシステムスタンダード)内蔵
 - ビデオインターフェイス(パラレル入力のプリンター)内蔵
- CPU部
 - CPU・トACOM MB8961使用(モトローラ6800相当)
 - オペレーティングシステム: モトローラM6800MIKBUG+CRTコントロール内蔵
 - I/O用ROM1K/CRT用RAM1K/ユーザー用RAM1K(model-01)
 - ユーザー用P1A×1 実装/ユーザー用ROM(ソケット付)
 - デジタルカセット・フロッピー等各種周辺機器接続容易
 - MIKBUG仕様ソフトウェア全て使用可/ S-W6800フルコンパチブルソフトウェア
 - P-ROMライター(MB8518用: インテル2706相当)実装
 - ユーザーエリア64Kまで内部拡張可能
 - 1K×8K BASIC・エディター・アセンブラー 各種MIKBUG仕様ライブラリーあり

WORLD WIDE
COMPUTER
SUPER SHOP

COSMOS[®]

アスターのSALES & CONSULTING WORK

マイコンおよびその周辺機器に関してあらゆる機種を扱っており、またあらゆるニーズに即応できる万全の態勢をいいております。そして、マイコンおよび周辺機器に関するコンサルティングワークも行なっておりますのでお気軽にご利用下さい。

好評発売中!!

カラーVIDEO RAM

本システムはLKIT-8もしくはMEK-6800D II等の6800系のCPU、8080系のCPUいずれのものでも接続できるVIDEO RAM MK-IIを使用して、カラーディスプレイを操作できるよう設計された低価格ビデオラムです。また、64字/行、32字/行の切換えやスクローリング、ページ転送の切換えもキーボードから操作できるようになっていますので非常に簡便な使用が可能となります。

また、LKIT-8及びMEK-6800D II両用のマザーボード(¥10,000)をご利用になると一層のシステムアップが実現できます。

LKIT-8システムアップ用RAMボード・ROMボード等も近日発表予定ですであわせてご利用下さい。

VIDEO RAM MK-IIはカラー信号用のためのRAMを内蔵したカラーキャラクタディスプレイです。

①文字7×9ドット、1行32 64文字切換え、1画面16行でアルファベット、数字及びカタカナが表示でき、色は7種類、文字カラー、バックカラー切換え
②6800系システム(MC6800、MB8661、HD46800、MC8502)タイミングコンパテブル

③回路を一層変更することにより8080系システムにも使用できます。(参考回路図付)

④色指定、32文字→64文字切換え可

VIDEO RAM MK-IIはCPUシステムのメモリの一部として使用する様になっていますので直接キーボードを接続して、CRTモニターとして使用することもできます。V-RAMをコントロールするための回路が必要となります。

⑤ライトペン回路外付可能

⑥ソフトウェアコントロールはCPUからのソフトウェアでコントロール

規格)

●電源 +5V 1.5A (max)
+12V 10mA (max)

●1画面 1024文字(64×16) 512文字(32×16)
ラスターキャン方式

●バス・ライン

データバス 8 Bit 双方向性正論理、TTLレベル
アドレスバス16Bit 9000番地迄に指定(正論理入力)

●ビデオ出力

周期、輝度、色信号、TTLレベル

テレビに接続するときは外部合成します。(参考回路図付)

●ライトペン用レジスタ端子付

レジスター及びライトペンを外付できます。(参考回路図付)

●1行 32文字、64文字切換え

切換えはアドレスを指定するだけで切換が出来ます。

●キャラクタジェネレーターはモトローラ社MCM6673AP使用

7×9ドットマトリクス、アルファベット、数字、カナ文字

●キャラクタジェネレーターをかけることにより、アルファベット小文字、ギリシア文字、コントロールマークなども表示可(MCM6671-6679)

最新入荷情報

- モトローラ PET2001 ¥298,000
- スイッチングレギュレーター 各々¥39,000
 - +5V10A±12V1A (01)
 - +5V10A±15V1A (02)
 - +5V10A±12V1A-5V1A (03)
 - +5V10A±15V1A-5V1A (04)
- 拡張用86PINエクササイザーコネクタ ¥1,900
(LKIT-8・MEK-DII etc)
- テクニクスバーサスター16bitマイコン ¥148,000
拡張システム完備
- コスモターミナルD用I/Oボード新発売(以下の規格)
 - ★コーサウHI/Oボード×6
 - ★デジタルセットインターフェイス
 - ★エクササイザーバスコンバーター
 - ★プリンター用インターフェイス
 - ★各ソフトウェア付

取扱品目(すべて即納態勢)

マイコン関係
ALTAIR6800
ALTAIR6800
APPLE-II
CHIBICOM
COSMAC
COSMOTERMINAL
EMIC S-100
F-8
H68TR
HEATH KIT各種
IMSAI 8048
IMSAI 8080
INTERCEPT JR
JOLT
KIM-1
LKIT-8
LKIT-16
MEK-6800D II
MELCS 8/2
MT60
NS SC/MIP
PET-2001
SDK-85
SOI-20

SOUTH WEST 6800
TECHNICO SUPER16
TK-80
TLC8-12A

周辺機器関係

リコーMIREADER
各種PTP、PTR
TEAC MT-2
TEAC MT-6
ADM-3A
ASR-33
SOUTH WEST CT-64
CRTディスプレイ各種
MS-6651(80characterプリンター)
SOUTH WEST PR-40
SP-12 超小型プリンター
IBM 725-735
VD-74C フロッピーディスク
Shugart フロッピーディスク
各種キーボード
各種リースバック端末周辺機器

新発売

コスモターミナル-D専用プリンター

●7×9ドットマトリクス ●80文字/行 ●カナ・英文字等128文字/行 ●100文字/行 ●グラフィック可能 ●ボールペン可変 ●文字サイズ可変 ¥295,000

札幌・仙台・東京・名古屋・大阪・鹿児島

アスターインターナショナル

株式会社 アスターインターナショナル
本社 東京都板橋区新板橋1-1-1 五層ビル
TEL 東京 03-3541 2461
札幌 011-531055 仙台 022-233-4050
名古屋 052-255 1505 福岡 092-233 2061
大阪 06-552 1400 東京 03-552 1400
大宮 048-552 1400 東京 03-552 1400
東京 03-552 1400 東京 03-552 1400
東京 03-552 1400 東京 03-552 1400

告 告 目 次

アスターインターナショナル	表2、1
コンピュータクラブ	6-7
ラジオ教育研究所	10
日立	20
三和無線測器研究所	21
NASA通信	22-23
関東バイトショップ	24
富士電子工業	25
ロビン電子産業	26
アドテック	27
若松通商	28
三真電機	29
大阪ICM	30
田中無線	31
東映無線	32
バックス/ノーズ	33
藤田電子	34-37
九管無線	38

アンツ	39
ダイデン商事	40
工人舎	41
I/Oラボラトリ	42
九十九電機	43
共立電子産業	44
サンベック	45
信越電気商会	46
ESDラボラトリ	49-52
東京スタンダード	2
大宮マイコンクラブ	47
テクニカルサンヨー	47
テックメイト	48
日本マイクロコンピュータ	48
ミズデンマイコンショップ	154
東芝	表3
日本電気	表4

マイコン

月販有り。別記の「月賦販売コーナー」を参照下さい。

SDK-85(インテル)キット	¥ 81,000 千サービス
TK-80BS(日電)端末	¥ 128,000
TK-80E(日電)キット	¥ 67,000
TK-80(日電)キット	¥ 87,000
MK-80A(インターナショナル)キット	¥ 68,000
TLCGS-12A-EX5(東芝)キット	¥ 77,000
H68 TR(日立)完成品	¥ 99,500
LKIT-8(富士通)完成品	¥ 85,000
SCMP(ナショナル・セミコン)キット	¥ 35,000
(ナショナル・セミコン)キーボード	¥ 38,500
LKIT-115(パナファコム)キット	¥ 98,000
MP-80(ロジック・システム)キット	¥ 39,500

端末 (送料実費)

TTY-ASR-89	¥ 540,000
カシオ・タイピュタ モデル501 TLTレベル	¥ 350,000
カシオ・タイピュタ モデル600型20A型	¥ 1,100,000
アリク・デューリター(TLTレベル・取り付き)	¥ 155,000
高調製・テープパンチャ(フォート800字毎分)PTCR32	¥ 19,000
高調製・テープパンチャ(1500字毎分)PTP-25	¥ 20,000

マイクロコンピュータ通信講座

マイコン 14講 別記の「月賦販売コーナー」を参照下さい。

●監修 東京大学教授 渡辺 茂 ●制作 マイテック

- 講座内容 各講、質問受付
1. マイコンコンピュータの基礎知識(1-5講)
 2. マイコンコンピュータシステム製作の実際(5-8講)
 3. マイコンコンピュータキットの製作(9-12講)
 4. マイコンコンピュータ開発と応用(13-14講)

●受講料 1名につき 33,000円 3名以上 32,000円

御注文は次の方法で①現金書留②電話③ハガキ④郵便為替⑤郵便振替(東京6-49308)但し②と③は代金引換扱いとなり実費が加算されます。 ●通販部 ●

東京スタンダード株式会社

〒145東京都大田区上池台3-25-3 TEL 東京03-727-8401

月賦販売コーナー

●下記の内、希望品名、回数を明記の上、申し込み下さい。(預金の有るものは、預金と共に申し込み下さい。) 送料込価格
●その他のマイコン・端末月賦有り。お問合せ下さい。

品 名	各回数	現金(前払)	月賦(後払)	支払合計
マイコン14講 マイテック 通信講座	2 3	13,000円 10,000円	10,000円 8,000円	33,000円 34,000円
TK-80E 日電 キット	4 5 10	4,000円 25,000円 0	7,700円 8,000円 7,700円	70,800円 73,000円 77,000円
H68 TR 日立 完成品	4 5 10 20	50,000円 25,000円 0 0	13,000円 10,200円 11,100円 5,180円	102,200円 106,600円 111,000円 122,800円
SDK-85 インテル キット	4 5 10 20	30,000円 25,000円 0 0	8,800円 8,200円 9,500円 5,300円	85,200円 90,600円 115,000円 108,000円
MK-80A インターナショナル インクエプティック キット	4 5 10 20	30,000円 20,000円 0 0	9,500円 8,300円 8,050円 5,700円	64,000円 69,800円 72,000円 81,500円
MEK8000DIIA モトローラ 完成品	4 5 10 20	40,000円 25,000円 0 0	10,200円 9,200円 9,250円 5,700円	80,800円 89,200円 92,500円 104,000円
TK-80BS 日電 端末	4 5 10 20	80,000円 50,000円 0 0	8,200円 8,300円 9,500円 7,600円	129,200円 133,500円 144,000円 152,000円
LKIT-11 パナファ キット	4 5 10 20	30,000円 25,000円 0 0	13,000円 10,600円 11,500円 5,700円	102,000円 109,800円 114,000円 124,000円
LKIT-8 富士通 キット	4 5 10 20	50,000円 25,000円 0 0	9,500円 11,100円 9,700円 5,370円	88,800円 91,800円 107,400円 107,450円
MP-80 ロジックシステム キット	2 3 4 10	20,000円 13,000円 10,000円 0	10,400円 9,300円 8,000円 4,500円	40,800円 41,500円 42,000円 44,000円
ASR-38 テレタイプ 輸入品	3 4 10 20	200,000円 200,000円 0 0	145,000円 74,000円 55,000円 37,000円	635,000円 844,000円 860,000円 740,000円

特集…マシン語に挑戦する?

全リスト公開

●マシン語に弱いあなたのための

TK-80用

ミニ・アセンブラ

中島義典……………53

TK-80

●ソフトウェア開発時に使って便利な

トレーサ

王村卓也……………64

LKIT-16

●他人の作ったプログラムもこれでバッチリ

逆アセンブラ

稲岡泰宏……………71

ワンチップ
CPU

INTEL 8086

Mr. 1 CHIP……………134

INTEL 8085

Mr. 1 CHIP……………136



送信

8080モルスの符号発生

藤 英一……………117

受信

H68/TRモルスの符号解説

大西義純……………120

●モ〜楽しいゲームです

TK-80BS MOOゲーム

村田 洋……………11

●君は何回で当るかな?

H68/TV-01数あてゲーム

北原 毅……………79

GAME

●美しいカラーでマイコンを

LKIT-16カラーキャラクタ・ディスプレイ

奥山昌男……………85

●4K BASICも1発スタート

H68/TRオートスタート・プログラム

北原 毅……………132

●+12V、+5V、-5V

マイコン用電源の製作

一條 博……………130

●ラボ・オートに最適

F/V、V/Fコンバータ

ハマヤ技研技術部……………155

実験

&

製作

Inside
Story

●アメリカ・マイコン業界も話と実際は……

マイコン旅行裏はなし

水島敏雄……………96

連

載

工業英語講座④〈MC6802〉

高木 敦……………98

Z-80マイクロコンピュータの製作④〈命令〉

東 光一……………113

BASICを始めよう〈初級〉③

佐藤雅春……………142

BASICで遊ぼう〈中級〉②

手塚佐知……………146

TK-80BS入門③〈BSのステートメント〉

戸塚文男……………149

キャラクタ・ディスプレイ・ターミナル③

インターフェース……………109

電子回路入門③〈電源〉

江口敏彦……………125

ミュージック・シンセサイザ徹底研究③

川島正裕……………15

〈新連載〉PALO ALTOの拡張①

東山マイコンクラブ……………101

買物ガイド

タウン情報

☆RANDOM BOX……………4,63

☆NEW PRODUCTS……………158

☆秋葉原/中京/日本橋マップ……………159

☆I/Oバザール……………100

☆BIG I/Oプラザ……………84

☆I/Oポート……………94

☆丸善洋書案内/ソフトウェア・サービス……………141

*イラスト=はらJIN+きむらしんじ

XTHL命令を使う

一條 博

8080系のマイコンの持つ毛色の変わった命令として、XTHL命令があります。これは便利な命令なのですが、どういうわけかあまりマイコン雑誌に解説がみえません。そこで、この命令の使用例を紹介しましょう。

XTHL

インテル社のカタログによると、

(L) \leftrightarrow (SP)
(H) \leftrightarrow ((SP) + 1)

Lレジスタの内容はSPレジスタで示されるアドレスの内容と交換され、Hレジスタの内容はSPレジスタで示されるアドレス+1の内容と交換されます。

このような動作の命令は、ライバルである6800にはない命令で、「知る人ぞ知る」命令なわけです。これを図解すると図1のようになり、XTHL命令では、αとLレジスタの内容、βとHレジスタの内容が交換されることになります。これだけでは使い方がないようには考えられますが……。

1. 6800対抗策

80が持っている命令はすべて有効に使い、少しでもライバルである6800より使いやすく、ということを使ってみます。

6800の持つ独特の命令で、SWI命令(Software interrupt)がありますがMIKBUGの管理のもとでプログラムのデバックを行なうとき非常に助かります。^{*}

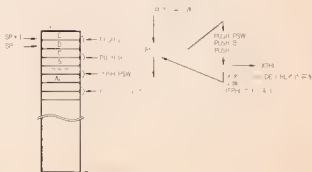
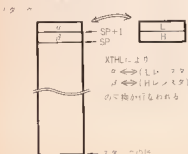
つまり、デバック中のプログラムのある所にSWIを入れておき、プログラムを実行するところの点までプログラムを実行して止めます。

以下、マニュアルの記述にしたがうと、

PCの内容が+1され次にPC、IX、AccA、AccB、CCレジスタがスタックに入れられる。

とあります。つまり、SWIでユーザ・プログラムの実行を中断し、その時の各レジスタの内容をセーブしているわけですから、スタックの内容を見ればプログラムの進行による各レジスタを確認できることになり、デバックがかなり容易になります。これはMIKBUG中にも有効に使われて

図1 XTHL命令



いるようです。

これに対応する命令を8080は持っていません。そこで、次のような方法が考えられます。つまり、6800においてSWIを入れる所を、8080では何らかのもどりアドレスをスタックに入れる分岐命令を入れます。

分岐命令としてはCALLなどがありますが、3バイトのため良い方法とはいえません。例えば、デバック後のプログラムの視察がめんどいことです。また、SWI命令も使いにくく、今回は1バイトの分岐命令であるRST命令を使用します。例えば、RST7で分岐すれば38H番地に強制的に飛び、スタックにはもどりのアドレスがセーブされます。ここで図2のようにPUSH PSW, R, D, Hなどを行なったXTHL命令を実行すればHLレジスタは、スタックの一番上の2バイトの内容と交換されます。

あとはスタックのポインタを操作しながらXTHLを使用すればCPU内のレジスタをすべて一度にしらべることが可能になります。(※ENGINEERING NOTE 100)

2. DATAのプリント

今のマイコンでは通常プログラムは原則としてROM、データはRAMに入れるのが通例ですが、その昔(今もそうなんだろうが)ミニコンのメモリとしてはコアを使ったものが多いというわけでもないでしょうが、プログラムの中にデータを入れてしまうことが多いようです。

それではというわけではないのですが自作のモニタなどで、何かいつも決まったコメントを出力することがあると思います。このような時、プログラムもコメントも同一のROMに入れてしまうと便利で、このような用途にもXTHL命令は使用できます。図3にその例を示しましょう。

PMESがプリントするサブルーチンでプリントされるデータはMESにあります。PMESにプログラムが移った後は、スタックの一番終わりにはサブルーチンよりのもどりアドレスとしてMESが残っています。

図2 全レジスタの内容を調べる

コンピュータ・ラブは
がんばります!

ご好評に応じて

Go! Go!

あのAPPLE IIが

たったの



コンピュータ・ラブでは去る5月1日より5月25日まで開店一周を記念して、Kick-Off Saleを断行し、マイコン・ファンの圧倒的ご支持を頂きました。誌面を借りてお礼申しあげます。ファンの皆様の「セール期間を延長して!」という熱いご希望にお応えして、大幅値下げを断行することに致しました。「華麗なるマイコン」APPLE IIをこのチャンスにぜひお買い求めください。

APPLE II 4 KRAM/8KROMシステム
店頭渡し ¥420,000
マニュアル付 20KRAM/8KROMシステム
デモテープ ¥540,000
(日本語取扱説明書付)

★APPLE IIをお買い求めの方に朗報!
★マイコン中古下取りセール実施中!
★TK-80もこの機会にぜひ!

別途付属品

- | | |
|--|---------|
| 1. キャリングケース | 12,000 |
| 2. RF モジュレータキット | 5,000 |
| 3. PR40Z プリンタキット 40字/行 | 99,800 |
| 完成品 | 130,000 |
| アップル用インターフェイス | 15,000 |
| 4. プロ用RQ-101Pドラムインパクト
プリンタ完成品ケース付
インターフェイス | 250,000 |
| 5. デジタルカセットMT-2(電源ケースナシ)
インターフェイス | 25,000 |
| 予価 | 298,000 |
| 6. ミニフロッピー (6月下旬) | 72,000 |
| 7. プリンタ用/パラレルインターフェイス (APPLE 製) | 25,000 |
| 8. シリアルインターフェイス | 32,000 |
| 9. I/Oボード 入力8bit, 出力24bit | 10,100 |
| 10. ユニバーサルボード (APPLE 製) | |

ソフトウェア

- | | |
|------------------------|--------|
| 1. 10K BASIC 日本語マニュアル付 | 15,000 |
| 2. ゲームカセット (5種入り) | 4,800 |
| 3. 高分解能グラフィックスバイオリズム | 3,000 |
| 4. チェックブック | 10,000 |
| 5. 16K スタートレック (強烈!) | 9,000 |
| 6. 高分解能グラフィックスデモ | 3,000 |
| 7. 各層発生 | 3,000 |

マニュアル等

- | | |
|-------------------------|-------|
| 日本語 6KBASICとモニタコマンド解説 | 1,500 |
| 6502ハードウェアマニュアル | 3,500 |
| 6502ソフトウェアマニュアル | 3,500 |
| ハイリゾリューショングラフィックス (日本語) | 600 |
| シリアルインターフェイス (日本語) | 300 |

■ 求む、技術者!

ESDで楽しく働きませんか?

[資格] 物理、化学、電子、電気の知識のある方 (新卒可)

[お問い合わせ先] ☎(03)816-3911 仁部(にへ)宛にどうぞ

LAB LETTERS ¥220(〒150)

●コンピュータ・ラブの新しい情報など
楽しい記事が満載!

セミナー会員募集中

EPROM 書込みサービス
1/16 EPROM 11,000 → 8,000 2708 + 15 → 2,000
2716 + 15 → 2,000
2732 + 15 → 2,000
2764 + 15 → 2,000
27128 + 15 → 2,000
27256 + 15 → 2,000
27512 + 15 → 2,000
2764 + 15 → 2,000
27128 + 15 → 2,000
27256 + 15 → 2,000
27512 + 15 → 2,000

大巾値下げ続行中!

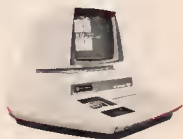
Sale

実施中!

¥ 420,000で!

PET200 I

¥ 298,000



8KRAM/14KROMシステム
CRTディスプレイ カセットレコーダ付

ESD LABはPETの各種テストを完了した結果、皆様におすすめることにしました。
モニタTV、オーディオカセットも組みこまれたシステムはあなたのPETです。

1. MICRO PROCESSOR
コンピュータラブで定評の6502

2. VIDEO DISPLAY
TEXT 8×8ドット 40字×25行(白黒反転可)
GRAPHICS 特殊なグラフィック専用文字(64字)使用で
図解を作り出せます。

3. MEMORY

RAM 内部最大8Kバイト、外部最大28Kバイト
ROM 8K BASIC+4Kモニタ+1K診断プログラム+1K
マシン語モニタ予定

4. BASIC

一般のBASICよりかなり高級です。10けた浮動小数点型
各種関数機能、PEEK、POKE、文字列

5. I/O部

IEEE-488入出力ポート
パラレルユーザボード 8bit
2台目のカセットインターフェイス

KIM-1 ¥69,800(完成品)
¥49,800(完成品)

- 2KモニタROM
- 1K RAM
- ZKEY/6デジ表示
- オーディオカセットインターフェイス/TTYインターフェイス
- 15プログラマブルI/Oポート
- プログラマブルクロック、割込み
- マニュアル、回路図付

★話題のVIM-1は6月テストを開始します!

- プログラムライター SHEPARDSON社 ¥397,000
- 704 2708 2715書込可能
- 505マイコン制御で命令は豊富です
- TTY RS232C I/Oポート付
- SCSI, BNF, BINARYデータOK

- LAB CRTディスプレイ ¥25,000
- 2枚1組回路図、説明書付
- LAB 8KローパワーRAMボードのみ(S-100) ¥60,000
- LAB 32KローパワーRAMボードのみ(S-100) ¥12,000(近日常売)

TECHNIC社T9900 スーパースターキット (16ビットチップ
同上用 32Kメモリキット 6月発売予定
同上用 192 キット 現在テスト中)

- CPU 6502 ¥7,500
- モニタ 6520 004 (TIM) ¥12,000
- PIA 6532 ¥7,500
- PIA 6522 ¥4,500
- メモリ 211C2 ロット ¥400

コンピュータ・ラブ チェーン



株イーエスディラボラトリ
〒113 東京都文京区本郷6-16-3
華伸ビル ☎ (03)816-3911

ラブ1
☎ (03)812-4911



ラブ2



堂々10,000部
圧倒的人気で発売中!

I/O別冊①

—マイコンづくりからBASICまで—

マイコン徹底研究

B5判 256頁 1,900円(〒200)

◆内容

- M6800マイコンのつくり方
- キャラクタ・ディスプレイのつくり方
- プリンタ、キーボード、A/D、D/A
とのインターフェイス
- フロッピーディスク・コントローラの製作
- 4K BASIC
- 宇宙船ゲームRACE
- MEK6800DⅡ, H68/TR, LKIT-8
- SWTPC CT-64
- SWTPC MP-68

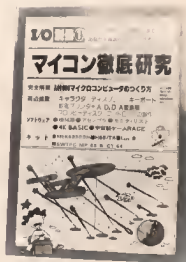


- 「マイコン徹底研究」のお買求めは、お近くの
屋さんか、I/Oが置いてある電気店へどうぞ。
- 「マイコン徹底研究」は大好評で売り切れ店が
続出しています。お近くに、本屋さんがない場合や
早急に本書を入手したい方は、直接工学社へお申
し込み下さい。

▶現金書留

〒151 東京都渋谷区代々木2-5-1 羽田ビル507
工学社営業部

▶郵便振替 〈東京5-22510〉



工 学 社

I/O別冊②

TTLからBASICゲームまで
あなたをゲームのエキスパートにする

TVゲーム

徹底研究

8080

マイコン・ゲーム

- ハノイの塔 ●絵かき ●迷路
- クリンゴン・キャブチャー
- コンピュータ学習機

TK-80BS

BASICゲーム

- カレンダー ●サインメくずし

TTL

魚雷船ゲーム

**TVゲーム用
LSI**

GI, MOS, NS, TI, 沖etc.

- 宇宙船 ●カーレース ●タンク戦争
- 各種ボールゲーム



■マイコンでTVゲームをつくろうというホビースト

■TTLでTVゲームをつくろうという元氣な人

■LSIで簡単にTVゲームを楽しもうという入門者

—— TVゲーム・ファン待望の手引書出来!! ——

総ページ224頁 定価1900円(送料200円)

既刊

I/O別冊①

■マイコン徹底研究

マイクロコンピュータの組み立てから、機械語、BASICまで
M6800を中心に、わかりやすく解説

定価1,900円(送料200円)

■I/O合本① [創刊号〜'77.2月号まで結集]

定価1,900円(送料160円)

■I/O合本② ['77.3月号〜5月号まで結集]

定価1,900円(送料160円)

お申込は、I/Oが置いてある

お店か、直接工学社へ

工 学 社

東京都渋谷区代々木2-5-1

羽田ビル507 ☎151

郵便振替東京5-22510

マイコン付 実習通信講座

完成品

最新鋭のマイコンが教材
の完全セット あなたが専
用のマイコンが持てる!!

▶当講座が開発したマイコン
RMC-1007 (電源付き)



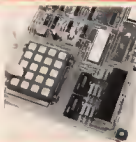
初めてでも
すぐ使える

マイコン

マイコンのすべてが学べるマイコン
付実習通信講座が誕生しました。

あなたの知識をもう一度整理して、
再確認するチャンスです。

初心者はもちろん、学生の方、プロ
の方にも、自信を持っておすすめでき
る通信講座です。



自宅学習で 完全マスター

専門家にも「わかりやすい」と好評
のテキストと、使いやすいマイコンシ
本体で、一流の先生方が、原理から応用
プログラミングまでマイコンのすべて
を指導します。

あなたも短期の自宅学習で、マイコ
ンのすべてが身につけられ、思いのま
まに使いこなせます。

RMC-1007は本格派マイコン

- | | |
|----------------------|-------------------|
| このマイコンRMC-1007は、 | 《ソフトウェア》 |
| ●基板サイズ 31cm×18cm | ROMにはキーボード、表示用コン |
| ●CPU 8080A ●メモリ: ROM | トロール用モニタプログラム、デー |
| 1Kバイト、RAM256バイト ●入力: | ジタルクロック、電子オルガン 等内 |
| 操作しやすいキーボード ●出力: 6 | 蔵 キーを数回操作するだけで、プロ |
| 桁セブンセグメント ●オーディオ: | グラムの知識のまったくない人でも、 |
| カセット入出力回路内蔵 ●I/O: | デジタルロック、電子オルガン等を |
| 8225使用 ★電源装置つき | 楽しむことができます。 |

詳しい案内書 無料急送

●ハガキに「マイコン」と明記して、
下記までご請求ください。詳しい案内
書を無料急送します。

〒167 東京都杉並区南荻窪4-29-10
TEL.03(334)0007

電子技術教育協会

10
5係

TK-80BS による

ムーゲーム

+CMT・インターフェイス



なにないマイコンクラブ

あるときは...C...電子専門学校 電子1学科 2 Aの学生
またあるときは...なにないマイコンクラブの発起と偏見によるボス
またあるときは...TK-80とTK-80BSをあやつる達人?
しかしその実態は...I/Oを愛する一匹狼なのだ

村田 洋

はじめに

マイコンキット (TK-80) は買ったけれどもなにがなんだかさっぱりわからず、結局付属のプログラム+I/O連数のプログラムなどを、マイコンにやらせて、はや半年、「俺みたいな凡人にはマシン語は難しすぎる／もっとやさしい(とI/Oに書いてあった) BASICでできやだめだ。」と、毎夜夢にまで見たBASICを、3月上旬友人がTK-BSを買うということに刺激されて、金もないのにBSを買った(〇〇ローンというやつで)

5日間寝マニアルとにらめっこして、「これなら俺にもプログラムができさうだ。」と思い、何かプログラムを作ろうと考えたのは6日目の夜。I/Oを創刊号から目を通してヒントになる記事は?とさがしたら...ありました! 1978, 1月号に「数あてゲーム」というのが、「よし、これにしよう。」と思って記事を読んだら、な、な、なんとこのゲーム、マシン語であるがTK-80を使っているではないか。「こんなおっさん(まだ若かったらゴメンナサイ、北原さん)に負けたまうか」と、またヒントをもとめて...すると「マイコンゲーム2!」という本に同じゲームが載っているではないか。さっそく2つの本を参考にしながらプログラムを作って...完結したのは8日目の夕方。初めてにしてはすばらしいと自分で感心し、あのおっさんを見返してやろうとI/Oに投稿した次第なのであります。

MOOゲームとは?

MOO (モーじやないよ、あちら(英国)の牛はム〜と

写真! CMT・インターフェイス基板

(下にLM565, 真中にMC14049, 565の右に741, 741の上はスチロールコンデンサ, 14049の右は発光ダイオード)基板上部のICソケットやC,Rは関係ない。

鳴くのです。ハイ)ゲームは英国で作られた数あてゲームで、コンピュータがひまぐれ?に選んだ4桁の数(MOO数)を、プレイヤーが当てるゲームです(ただし4桁の数は同じ数字を2つ以上含まない)。入力した推測数に対してコンピュータは、あるヒントを出します。そのヒントというのは、推測数の中にMOO数と同じ数があればメウシ、桁まで同じならオウシを表示します。たとえば、MOO=1234、推測数=1428のときには、オウシ1頭、メウシ2頭というぐあいです。数回の推理をくり返して表示をオウシ4頭(この時MOO=推測数)にするというゲームです。

プログラム説明

DAN

行10~110は、ゲーム説明ですので省略しても結構です。(人に見せるときは、入れておいた方がよいと思いますが)
行120~230は、乱数によるMOO数の作成です。B, C, Dに0が少々出やすいのですがゲームをする分には影響ないと思います。
行240~330は、推測数の代入とその数を1桁ずつ分解します。最後の行330はウシをディスプレイするためのカーソルの初期値です。
行340~510は、MOO数と推測数の比較とその結果の表示、正解かどうか調べます。
行520~870は、省略します。
行2000~は、タイマーのサブルーチン
行3000~は、オウシのディスプレイ用のサブルーチン
行5000~は、メウシのディスプレイ用のサブルーチン

図1 オウシの例

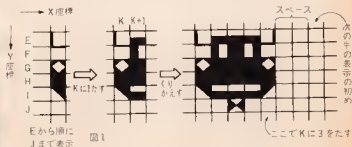


写真2 1回目の推測数を入力したところ

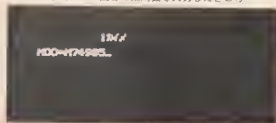


写真4 2回目の推測数を入力

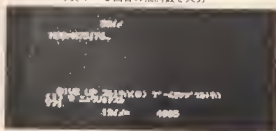


写真3 残念ながら牛は0頭(4905はハズレ)

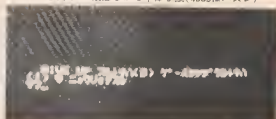
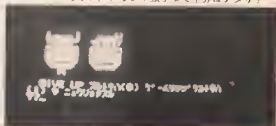


写真5 オウシ・メウシ1頭ずつで半分はアタリ?



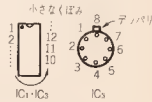
サブルーチン

このプログラムで、もっとも苦労したのが、ウシのサブルーチンです。どうしてもウシを最高4頭まで表示させられるか、咄い頭からひねりにひねり出したところまで。

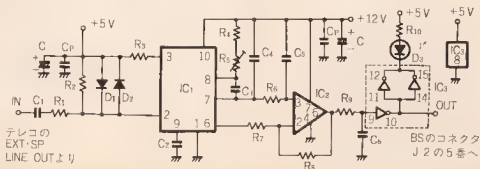
このゲームでは、ウシは左から右へ縦に1行ずつ表示させていきます。いまX座標をKとすると、座標Kの縦の列を表示し終えると次にKに1を足せば右どなりの列に移ることができます。そしてK+1の縦の列を表示すると、また1を足す。これをくり返して10回分表示して最後に次の1とのスペースのためにKに3を足します。(図1参照)

図2 PLL・ICを用いたDMT・インターフェイス

R ₁	5.6K	C ₁	0.01μ(セラミック)
R ₂	1K	C ₂	0.047μ(マイラ)
R ₃	1K	C ₃	1000p(スチコン)
R ₄	1K	C ₄	0.047μ(セラミック)
R ₅	5K	C ₅	0.047μ(")
R ₆	5.6K	C ₆	0.01μ(")
R ₇	1K	D ₁ , D ₂	1S1588(SW用ならどれて可)
R ₈	1M	D ₃	LED
R ₉	10K	IC ₁	LM565(565ならどれても)
R ₁₀	220Ω	IC ₂	μA741(741ならどれても)
C _P	0.01μ(セラミック)	IC ₃	MC14049(μPD4049可)
C	22μ 16V		



上面図



カセット・インターフェイス

T K-80BSをお持ちの方でカセット・インターフェイスで困っておられる方が少なくないと思います。(俺も友人もその1人でした。過去形) 短いときはなんとか入ってくれるのだが、ちょっと長くなるともうダメ。MOOゲームのときなんか30分かってやっとロードできた仕木。(次の日は1時間かかっても入らずじまい。タタリじゃー) テレコを褒えたり、デ・キでやってみたりしたがこれもダメ。「これは回路がダメなんだ。」と、勝手に決め込んで、さっそくPLL・IC使用の回路を自作して、それでやってみたら、今までエラーばかりしていたテープが一発でスココノと入ってしまった。それに再生レベルをある一定以上ならロード中に手で回したってまったくエラーしない。(6-7年前のテレコ+6巻1,000円の特価テープを使用してもいまだエラーなし) さすがPLL・ICの威力ノと友人と2人で感心しています。(このイン

小さくぼみ

写真6 オウシ4頭となりアタリ!



写真7 結果の表示

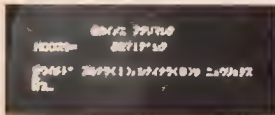


写真8 ゲーム終了の表示

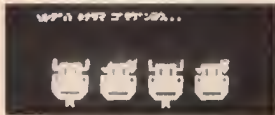
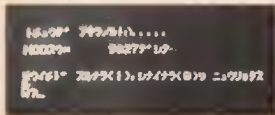


写真9 GIVE UPしたときの表示



ターフェイスにしてから、ロード途中でも安心してトイレに行っております。)

回路は図2を見ておわりのとおり非常に簡単に、かつ調整箇所も1箇所(ボリューム)だけです。(この調整も適当でよく、ただ信号を入力中に発光ダイオードを亮るようにすればよいだけです。)

TK-80BSとの接続は、図3(a)と図3(b)がありますがどちらでも結構です。(他はa、友人はbだが2人ともエラーなし。)

最後に

なにしろ簡単とはいえ、慣れない言語(他は語学に弱いので)BASICなるものを使って、それも十分に練習もしないうちにこのゲームを作ったので、ダラダラと長いものになってしまいました。後で考えてみると、もう少し短くできるんじゃないかなと思うところが数箇所でてきました。(修正するのが面倒なので作り変えていない)MOOゲームをやってみた感想は、思ったより当てるのが難しいということです。

図3(a) BSとのインターフェイス例

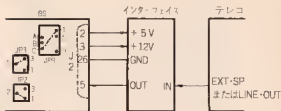


図3(a)

図3(b) BSとのインターフェイス例

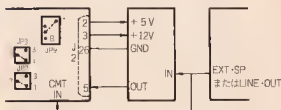
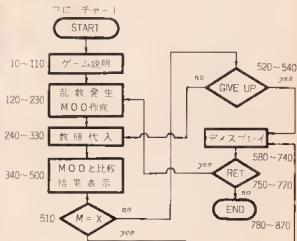


図3(b)



推測数が1回前のしか表示されないところに原因?があるようです。(今までの最高記録は2回、最低は25回)読者の方々も1回で当ててことを夢見て、MOOゲームを楽しんで下さい。また、我「なにおマイコンクラブ」では、BSによるゲームをいろいろ開発中ですので、順次発表していくつもり(連載のうわさも内部に流れております)です。 END

フローチャート



SWEET
TIME



I/Oプラザ

▶5月号のアメリカ・マイコン情報をみてびびり、なんと68000用のBASICコンパイラなるものが出まわっているではないか? それにひきかえ日本ではBASICインタープリタがやっと普及しはじめたところ。しかも、どうも動くベシヤ、あーんたる格差。そこでI/Oの使命だが、BASICコンパイラのリストを 安全公開せよ。

(西表島のカンピラ君より)

MOO(ムー)ゲーム・プログラム

```

10 P. "コレハ MOOゲーム デス"; GOS.2000
20 P. "カクダガ スベテ コトナリ 4ケタノ カズ(MOOスウ)ヲ
   アテルゲーム デス"; GOS.2000
30 P. "ニュウリョクスウノ アルケタガ MOOスウノ ソノケタノ
   カズト オナジナラ オウシヲ"; GOS.2000
40 P. "アルケタノカズガ MOOスウノ ナカニ アレバ メウシヲ
   ディスプレイ シマス"; GOS.2000
50 P. "アサヲハ ウシノヒントヲ タヨリニ カズヲ アテルデス"; GOS.2000
60 P. "ニュウリョクシタカズト MOOスウガ オナジトナリタトキ
   スベテ オウシト ナリマス"; GOS.2000
70 P. "イカイ ニュウリョクシタカズハ モウイチド ニュウリョク
   シテハ イケマセン PLAYER ノ マナーデス"; GOS.2000
80 P. "OK?"; GOS.2000
90 P. "1ヲニュウリョクシテバ ゲームカインデス"
100 1.N.Z
110 IF Z#1 G.90
120 CL.
130 A=R(9)
140 B=R(10)-1
150 C=R(10)-1
160 D=R(10)-1
170 IF A=B G.140
180 IF A=C G.150
190 IF B=C G.150
200 IF A=D G.160
210 IF B=D G.160
220 IF C=D G.160
230 X=A*1000+B*100+C*10+D
240 L=1
250 CU.3,1;P.L,"カイメ"
260 P;
270 1.N "MOO"=,M
280 IF M<1000 G.270
290 O=M/1000
300 P=(M-O*1000)/100
310 Q=(M-O*1000-P*100)/10
320 R=M-O*1000-P*100-Q*10
330 E=7, F=8, G=9, H=10, I=11, J=12, K=4
340 CL.
350 IF A=O GOS.3000
360 IF B=P GOS.3000
370 IF C=Q GOS.3000
380 IF O=R GOS.3000
390 IF A=P GOS.5000
400 IF A=Q GOS.5000
410 IF A=R GOS.5000
420 IF B=O GOS.5000
430 IF B=Q GOS.5000
440 IF B=R GOS.5000
450 IF C=O GOS.5000
460 IF C=P GOS.5000
470 IF C=R GOS.5000
480 IF O=Q GOS.5000
490 IF D=P GOS.5000
500 IF D=Q GOS.5000
510 IF X=M G.580
520 CU.1,14;P."GIVE UP スルトキハ(0) ゲームヲソノケル
   トキハ(1)ヲ ニュウリョクスル"
530 1.N.Y
540 IF Y=0 G.690
550 CU.3,16;P.L,"カイメ"=,M
560 L=L+1
570 G.250
580 GOS.2000
590 CL.

```



```

600 IF L=1 G.640
610 IF L=2 G.650
620 IF L>9 G.660
630 P.L,"カイメニ アタリ マシタ";G.710
640 P."1カイメヲ アテルトハ、アサヲハ テンサイデス";G.710
650 P."2カイメヲ アテルトハ、アサヲハ シュウサイデス";G.710
660 P.L,"カイメ カカルトハ....."
670 P.;
680 P."カオヲ アラツテ デナオシテキテ クダサイ";G.710
690 CL.
700 P."トチュエデ アキラメルトハ....."
710 P.;
720 P."MOO"=,X,"デシタ"
730 P.;
740 P.;
750 P."モウイチド スルナラ(1) シナイナラ(0)ヲ ニュウリョク"
760 1.N.W
770 IF W=1 G.120
780 CL.
790 P."ソレデハ ミナサマ ゴキゲンヨウ"
800 K=4
810 GOS.3000
820 GOS.5000
830 GOS.3500
840 GOS.5000
850 GOS.2000
860 CL.
870 S.
2000 FOR T=1 TO 3000
2100 NE.T
2200 R.
3000 CU,K,E;P.H84;CU,K,F;P.H92;CU,K,G;P.HCB;
   CU,K,H;P.H80;CU,K,I;P.HB5
3100 K=K+1
3200 CU,K,E;P.H91;CU,K,F;P.H82;CU,K,G;P.H80;
   CU,K,H;P.H80;CU,K,I;P.H91
3300 K=K+1
3400 CU,K,E;P.H91;CU,K,F;P.H80;CU,K,G;P.H80;
   CU,K,H;P.H80;CU,K,I;P.H91
3450 CU,K,J;P.HB7
3500 K=K+1
3600 CU,K,E;P.H91;CU,K,F;P.H83;CU,K,G;P.H80;
   CU,K,H;P.H80;CU,K,I;P.H91
3700 K=K+1
3800 CU,K,E;P.H87;CU,K,F;P.H92;CU,K,G;P.HCB;
   CU,K,H;P.H80;CU,K,I;P.HB1
3900 K=K+3
4000 R.
5000 CU,K,F;P.HBB;CU,K,G;P.HCB;CU,K,H;P.H80;
   CU,K,I;P.HB5
5100 K=K+1
5200 CU,K,E;P.H91;CU,K,F;P.H82;CU,K,G;P.H80;
   CU,K,H;P.H80;CU,K,I;P.H91
5300 K=K+1
5400 CU,K,E;P.H91;CU,K,F;P.H80;CU,K,G;P.H80;
   CU,K,H;P.H80;CU,K,I;P.H91
5500 K=K+1
5600 CU,K,E;P.H91;CU,K,F;P.H83;CU,K,G;P.H80;
   CU,K,H;P.H80;CU,K,I;P.H91
5700 K=K+1
5800 CU,K,E;P.HB7;CU,K,F;P.HBB;CU,K,G;P.HCB;
   CU,K,H;P.H80;CU,K,I;P.HB1
5900 K=K+3
6000 R

```

▶前掲 SC/MP-IIは無い、最近、6800が同位にくらい、8080が同位くらいの値段で売られているが、SC/MP-IIはクロック・ジェネレータ内蔵で、SC/MP-IIはOMAやマルチCPUシステムが簡単に制御できる。めんどうなクロックがない。(加)コントロールバス、データバス、アドレスバスは3 STATE。また、汎用FLAG出力が3つもある。これは便利。SC/MP-IIは命令体系がSimple、6800なんて目じゃない。だから理解しやすい。SC/MP-IIのアドレッシングは相対を中心としている。(高級?)インデックス・レジスタが3つもあり

ミュージック・シンセサイザ

徹底研究 3



ローランド・システム100

川島 正裕

前回までに音のもとを作る音源、音源で発生した音の音色を変化させる各種フィルタ、音の出かたを制御するアンプ(VCA)について説明しましたが、これらの各種モジュールはほとんどが電圧制御形であるために、各モジュールをコントロールするコントロール電圧が必要になります。

今回は、色々なコントロール電圧を作り出す各種モジュールについて説明します。

音階の作り方



音の高さは、音源で作り出される音の発振周波数で決まります。ミュージック・シンセサイザでは、音源として、コントロール電圧で発振周波数の変化するVCOが使われているので、コントロール電圧を段階的に変化させれば、容易に音階が作り出せます。

一般に、VCOでは、1ボルト電圧を上げれば1オクターブ高い音になるので、12平均律の音階を得たい場合、1/12Vずつ変化させれば良いことになります。

ですから、図1のようにボリュームの両端に電圧を

写真1 システム700の鍵盤(左端に、ピッチ、ポルタメント、ベンダーが付いている)



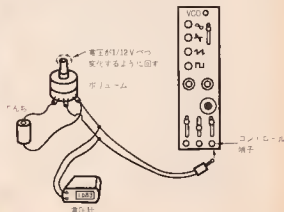
加えておいて、ボリュームのセンター端子からの電圧を、VCOのコントロール端子に加え、電圧計をにらみながら1/12Vずつ電圧を変化させてゆけば音階が得られるのですが、これではとても演奏することはできません。

そこで、普通のミュージック・シンセサイザでは、鍵盤によって音階を作るためのコントロール電圧を作り出しています。もちろん鍵盤以外のギターなどのようなものでも、1オクターブで1V変化する電圧を得られれば良いのですが、構造などの点で鍵盤がもっとも作り易いようです。

鍵盤(Key Board)

シンセサイザのモジュールとしての鍵盤には、最低2つの出力が必要となります。1つは、音階を作るためのコントロール電圧で、一般に鍵盤から得られるコントロール電圧という意味で、**Key Board Control Voltage**略して**KCV**と呼ばれます。もう1つは、鍵

図1 VCOによる音階の作り方



る。もちろん絶対アドレスシフト可能(インデックス・レジスタを0にしておく)これはすごい!「オート・インデックス」こんなアドレスシフトはZ-80にもないだろう!ワハハハハハハハ。SC/MP-IIには、BASIC(4K)が入ったROMが用意されている。こんなすばらしいCPUはない!みなさん!SC/MP-IIです。大SC/MP-IIバンザーイ!(俺はぜったいSC/MP派より)

図2 鍵盤のブロック図

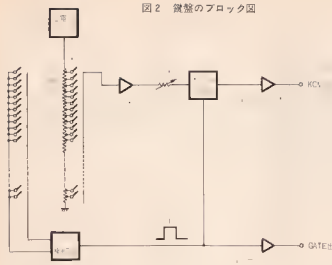
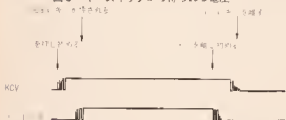


図3 キースイッチから得られる電圧



鍵盤が押されているかどうかを知らせる。ゲート信号 (Gate) で、これは単なる ON/OFF 信号です。

KCV は単音シンセサイザでは、1 つしか出ないわけ、もし鍵盤を 2 鍵以上押した場合は、押されているキーの中で最も低い音か、最も高い音に対応した KCV が得られます。前者を低音優先、後者を高音優先と呼び、どちらの方式を採用しているかは、シンセサイザによって異なりますが、シンセサイザを単音楽器として見た場合には、どちらの方式であっても音楽上支障はないと思います。

アープ・オデッセイ、ローランド・SH-7、システム 700 などの 2 ノートタイプのシンセサイザでは、高音優先と低音優先の 2 つの KCV が出ているので、1 キーだけしか押さなかった場合は、どちらの KCV も同じ電圧が出ているのですが、2 キー以上押した場合、押されたキーの中で最も低い音と、最も高い音に対応した KCV がそれぞれ得られます。

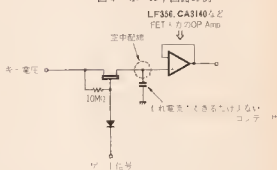
この 2 つの KCV をそれぞれ別の VCO に入れてやれば複音の出せるシンセサイザが作れるのですが、ゲート信号は 1 つしか得られないので、エンベロープの作り方などには制約があります。

図 2 に基本的な鍵盤回路のブロック図を示します。オームの法則により、キースイッチに接続された精密抵抗と定電流回路で $1V/OCT$ の KCV が得られます。この電圧をバッファ・アンプを通してそのまま VCO

図5 ボルトメント効果



図4 ホールド回路の例



のコントロール電圧として使えば、一応音階ができるのですが、キーを離すとすぐに電圧が下がってしまいます。そのため、キーを押している間だけしか音を出ない場合は良いのですが、キーを離してからでも鳴り続けるようなエンベロープにしたとき、都合の悪いことになります。そこでキーを離しても、いちばん最後に押されたキーに対応する電圧を保持し続けるように、アナログのホールド回路が入っています。

つまり、ゲート信号が ON の状態 (キーが押されている) のとき、ホールド回路には KCV が読み込まれ、ゲート信号が OFF の状態になれば、ホールド回路を閉じて前の KCV を保持し続けます。実際にキースイッチから得られる電圧は、図 3 に示すように、チャタリングを含んでいるので、KCV の方のキースイッチを、ゲート用のキースイッチより先に ON させ、KCV が充分安定した所でゲート信号を ON させて、ホールド回路に読み込ませます。また、キーを離すときも逆に、ゲート信号を先に OFF させています。

図 4 にホールド回路の例を示します。図からも分かるように、コンデンサの漏れ電流、オペアンプのオフセット電流などがあると、ホールドされた電圧は次第に変化して行きます。この変化の少ないものほど、優秀な鍵盤といえるのですが、やはりキーを押した状態

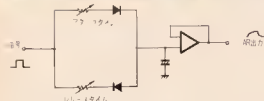
1/0 プラザ

▶ 詳報 はじめてお手紙を出します。ただ今、SC-MP-II の NR によるマイコンを作製中ですが、大変なことを発見いたしましたので買回せたい。SC-MP-II の NR 5T、CONT 端子がアクティブの場合データおよびアドレス・バスは HI-Z なのでは? かしこい。大変なことです。幸甚幸甚を願います。(NR 5T が

写真2 ローランドシステム700のエンベロープ・ジェネレータ(ADSR)



図7 ARの回路例



で最も安定したKCVが得られますから、チューニングをとるときは、必ずキーを押しながら行なうようにします。

鍵盤でもう1つ重要なものにポルタメントがあります。図5(a)に示すように、異なった鍵盤に押しかえたとき、KCVは瞬時に変化しますが、バイオリン、トロンボーン、口笛などのような音程がゆるやかに変化して行く、ポルタメントの効果を出すには、図5(b)のようにKCVをゆるやかに変化させる必要があります。そこで、ホールド回路の前に直列に可変の抵抗を入れて、ホールド回路のコンデンサとで積分回路を作っています。

この他に、鍵盤にはチューニングをとるためのピッチ・コントロール、鍵盤の音程を上下にオクターブ関係ですらすトランスポーズ(KCVの絶対値を1Vずらす)、レバーでKCVを変化させるベンダーなどが付いています。

エンベロープ・ジェネレータ

先月号で、エンベロープ(音の出方、消え方)を制御するために、VCAを使うと説明しましたが、実際には、VCAの増幅度を時々刻々変化させるための、コントロール電圧が必要となります。これを作るのがエンベロープ・ジェネレータで、一般にAR、ADSRと呼ばれるものが使われます。

図6 ARの動作

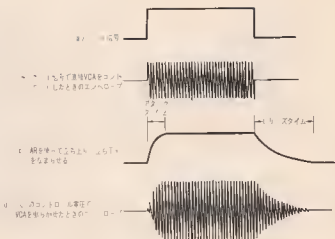
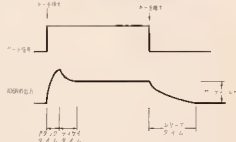


図8 ADSRの動作



●AR

ARのAは立ち上がり時間(Attack Time)、Rは立ち下り時間(Release Time)のことで、立ち上がり時間、立ち下り時間だけが変えられるエンベロープ・ジェネレータです。

鍵盤から出てくるゲート信号は、図6(a)のように単なるON-OFF信号ですから、これをそのまま、VCAのコントロール電圧として使ったのでは、図6(b)のように楽音信号を直接スイッチで断続したのと変わらず面白くありません。そこでARによって図6(c)のような立ち上り、立ち下りのなまった電圧で、VCAをコントロールしてやれば、図6(d)に示すように自然な感じのエンベロープになります。図7に示すようにARは、回路が簡単なために、簡易形のエンベロープ・ジェネレータとして使われます。

●ADSR

ADSRは、ARよりさらに変化に富んだエンベロープを作るために、アタック・タイム、リリース・タイムのほかに、ディケイ・タイム(Decay Time) サスティン・レベル(Sustain Level)が設定できるようになっています。

図8に示すように、ゲート信号がONになると、アタック・タイムで設定された時間かかって、最大電圧

*Lの時、CONTが「L」の時)どうかご返事ください。(東京都 下山智明) (SC/MP-IIでは、R F SETをかけた状態、つまりNRSTが「L」のときはH Zになりますが、CONT端子ではありません。一編)

写真3 ローランドシステム700のLFOとその内部

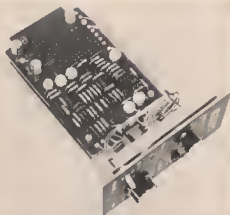
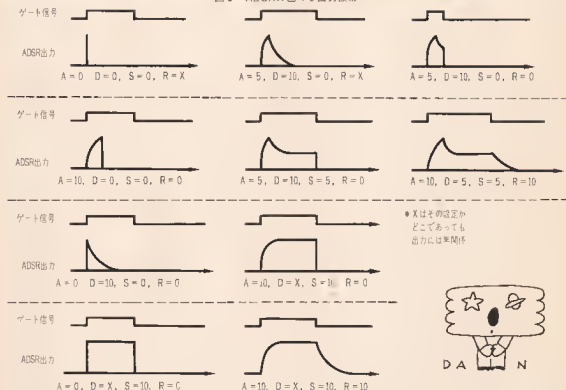


図9 ADSRの色々な出力波形



(普通10—15Vぐらい)に達し、次にディケイ・タイムで設定された時間かかって、サステイン・レベルで決められた電圧まで下がってきます。

サステイン・レベルが0に設定してあると、0Vまで電圧が下がるので、キーを押し続けても音は減衰して行きます。また、サステイン・レベルが最大にしてあると、ディケイ・タイムは無関係になります。キーを離すと、リリース・タイムで設定した時間かかって0Vまで下がります。サステイン・レベルが0で、キーを離す前に0Vまで下がっていた場合はリリース・タ

イムは無関係になります。

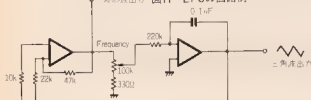
図9にADSRを色々な設定した場合の、出力電圧の変化を示しましたが、同じ設定であっても、キーを押している時間(ゲート信号がONの時間)によって出力電圧の変化カーブがちがってくる場合があります。A D S Rの設定と演奏方法とは、合わせて考える必要があります。

また、シンセサイザを演奏する場合、レガート(次のキーに移るとき、前のキーを離す前に、次のキーを押す奏法)に弾くと、ゲート信号は出っぱなしになり、

図10 トリガ信号のはたらき



図11 LFOの回路例



新しくキーが押されたかどうか、判断できなくなります。ですから、音程は変化しても、エンベロープに変化はつきません。キーを押し変えるごとにエンベロープ・ジェネレータを初めの部分から動かさせるには、ノン・レガート奏法が、スタッカート奏法（キーを短かく叩くように弾く）にする必要があります。逆に、この3つの演奏方法によって、演奏に変化を付けることができます。

ローランドSH-7、システム700などでは、鍵盤からゲート信号のほかに、トリガ信号（Trigger）が出ています。これは、新しくキーが押されてKCVが変化すること、短かいパルス信号が得られるもので、このトリガ信号により、エンベロープ・ジェネレータを再トリガすれば、レガートに弾いても、音程が変化すること、エンベロープの初めの部分から動作することになります（図10）。

エンベロープ・ジェネレータの出力電圧をVCAのコントロール電圧として使えば、音のエンベロープをコントロールできますが、このほかに、VCFやVCOのコントロール電圧として使えば、時々刻々、音色や音程が変化して行き、面白い効果が得られます。

【カラーテレビは、赤、青、緑の加色混合（3つの色を混ぜると白くなる）を用いています。上井谷君は絵の具などで用いられる、減色混合（3つの色を混ぜると黒くなる）と勘違いをしているのか？・・・Laser-16でもいろいろプログラムができたら、I/Oに違ってホ、一編一】

ローランドの新しいシンセサイザSH-1



図12 LFOの鍵盤によるトリガ

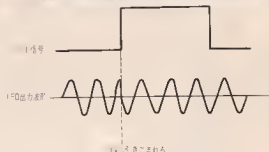
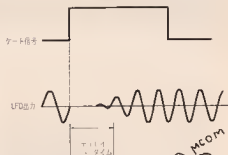


図13 LFOのディレイ効果



LFO



Low Frequency Oscillatorの略で、変調用の低い周波数（0.01～20Hzぐらい）の発振器です。周波数の可変範囲を大きくするため、図11に示すような、コンパレータと、積分器を組み合わせた物がよく使われるようです。

LFOを常に発振しているのですが、場合によってキーに連動して0Vの点から波形を出したいことがあるので、ゲート信号や、トリガ信号によって強制的に波形を0Vに引き込むことができるようになっていきます（図12）また、バイオリンなどの演奏で聞かれる、ディレイやビブラートの感じを出すために、鍵盤からのゲート信号に同期して、LFOの出力を少しの間、止める回路も入っています（図13）また高純なLFOでは発振周波数を、電圧でコントロールできるVCOタイプのものもあります。



家庭用テレビやモニタテレビが高分解能のグラフィックディスプレイになります。

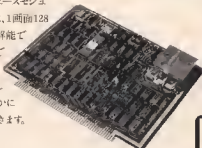
新しく発売されたテレビインタフェースモジュールH68/TVは、トレーニングモジュールH68/TRと接続してテレビ画面を制御するモジュールです。家庭用テレビやモニタテレビを簡単にキャラクターディスプレイあるいはグラフィックディスプレイに変えてしまいます。このテレビインタフェースモジュールの最大の長は、1画面128×96ドットという高分解能でグラフィック表示ができるという点にあります。そのため表示した図形をかなり滑らかに移動させることができます。

＜H68/TVの特長＞

■文字表示モードでは、512字(32字×16行)または、1,024字(64字×16行)を1画面に表示できます。



※1,024字は、モニタテレビ使用の場合に可能



■グラフィックモードでは、1画面128×96ドットで、動画を表示することができます。



- 漢字、片仮名、ゲーム用図形など234種類のパターンをサブルーチンとして用意しています。(グラフィックモード)
- カーソル表示、スクロール表示、ブランク行、ページングの画面制御機能があります。
- ソフトウェアによって白黒反転が可能です。
- 全英語・日本語BASICも使えます。
- モニタ、サブルーチンを提供します。

＜仕 様＞

表示モード ●文字表示モード、グラフィックモード

文字種類 ●標準64種

文字、図形サイズ ●文字:15×7ドット

図形:8×12ドット

図形種類 ●グラフィックモード:234種
(サブルーチンによって提供)

●文字表示モード:32種

(オプション)

アセンブリでプログラミングできる:
トレーニングモジュールH68/TR



テレビゲームから、オンラインコントロールの実験まで、フルキーボードの専用コンソール、オーディオカセットテープレコーダ、タイマ相込み機能などを生かして、さまざまな使い方ができます。

日立テレビインタフェースモジュール

H68/TV — ¥69,500

P-ROM ライター & チェッカー

Model SPW-4010

新製品

Model SPW-4010は、次のような数々の機能を備えているP-ROMライター&チェッカーです。価格も驚く程低価格におさえてありますから、どなたにもご使用いただけます。

■ P-ROMの品種 2704 / 2708および2758 / 2716の2組…どちらかの1組用ボードはオプション販売です。

■ 書き込み、ペリファイ、チェック機能 P-ROMの書き込み器として充分な機能を発揮する他、さらに書き込み後のペリファイおよび出力電圧をチェックすることができます。

■ 外部コントロール機能 本器はマニュアル操作型で、次の理由からCPUを内蔵することを避けております。CPU、PTR、TTY等からの外部信号でデバックされた命令コードをP-ROMに書き込むことができ、



これも本器の特長となっています。…CPU、PTR、TTY等のインターフェース・ボードはオプション販売。

■ コピー機能 すでに書き込みの済んだP-ROMからのプログラムコードを、新しいP-ROMにコピーできます。

■ リスト機能 パネル上のLEDに、アド

レスとデータの表示ができます。

■ 消去チェック機能 消去のチェックができます。

μPD-454D、1702その他のP-ROM用のものもご希望により製作いたします。

P-ROM イレーサー

Model SPE-4020

新製品

Model SPE-4020 P-ROM イレーサーは、どなたにも簡単にP-ROMの消去が行え、価格も画期的な低価格ですから手軽にご使用願えます。

消去を行うには台の上にP-ROMを置き、その上に本器をかぶせて電源スイッチ、スタートボタンをONにするだけで動作を開始します。誤って紫外線を直視する事を防ぐ

ために、手に持ったままでランプを見ながら電源スイッチを“ON”にしても、直ちに点灯することなく、更にスタートボタンを押した上、ケース内部の自動スイッチを操作するまでは動作しないようになっています。操作容易、低価格そして安全設計と3拍子揃ったサンワP-ROM イレーサーを業務



用に、またホビー用にも是非ご愛用ください。

SANWA

SANWA RADIO MEASUREMENT WORKS
三和無線測器研究所

〔本社・工場〕東京都国分寺市東恋ヶ窪4-29-4 TEL.0423(25)3030(代)

可能性への挑戦。



計算するライター

ICカキコイター

- ゴールド(金) ¥15,000
- ブラック(黒) ¥12,000
- シルバー(銀) ¥10,000
(無手数料)

代理店募集

OEMの方は、価格をご相談
ください。

オフィスコンピュータ・マイクロコンピュータ・電子パーツ・
美観無線・システム情報機器・関係書籍

NASAマイコン

NASAコンピュータ事業部

事業部 ● 甲府市塩部一丁目9-10 ☎(0552)53-7373代

本社 ● 甲府市丸の内一丁目9-19 県信ビル 工場 ● 山梨県中巨摩郡竜王町名取370-4
☎(0552)37-7373代 ☎(05527)6-7373代

山梨マイコンクラブ 会員募集中

会費 1000円
送料 送料別



厳選された精鋭たち



●東芝 TLCS-80A-EX-80
¥85,000 テーサービス



●NEC TK80BS
¥128,000
テーサービス
TK80.80E用
BASIC KIT



●NEC TK80E
¥67,000 テーサービス



●日立 H68/TR
¥99,500 テーサービス



●パナファコム L-KIT16
¥98,000 テーサービス



●ナショナルパナキットKX-33B
¥39,800 テーサービス



●TSP7706A
放電プリンター
(インターフェース
内蔵)
¥37,000
テーサービス

●NASAプログラム用
カセット (ROBIN C-60) ¥200
テープ (NASA C-60) ¥300

●松久キーボード ¥70,000



エンコードなし ¥18,000

TDK

TRM020シリーズ ¥35,000 ¥1,000



TRM021	5V 5A	+12V (0.3A)	12V (0.3A)
TRM022	5V 5A	+15V (0.3A)	15V (0.3A)
TRM023	1.5V 1.5A	+12V (0.5A)	12V (0.5A)
TRM024	5V 1.5A	+15V (0.3A)	15V (0.3A)
TRM025	5V 1.5A	+12V (0.3A)	12V (0.3A)

TRM000シリーズ ¥49,000 ¥1,000



TAM-001	5V 30A	+12V (1A)	-12V (1A)
TDM-002	5V 10A	+15V (1A)	15V (1A)
TRM-003	5V 10A	+12V (1A)	12V (1A)
TAM-004	5V 30A	+15V (1A)	15V (1A)
TRM-005	5V 30A	+12V (1A)	12V (1A)

TDK

RMシリーズ ¥25,000 ¥1,000



RM01-005	1.5V 0.6A	4.5-5.5V H/L
RM05-005	5V 0.2A	5.0-10.0V H/L
RM10-005	12V 0.2A	10.0-13.5V H/L
RM15-005	15V 0.2A	12.5-16.5V H/L
RM24-005	24V 0.2A	21.0-26.5V H/L

RMシリーズ

¥31,400 ¥1,000



RM05-105	5V 10A	4.5-5.5V H/L
RM05-005	5V 0.2A	5.0-10.0V H/L
RM10-005	12V 0.2A	10.0-13.5V H/L
RM15-005	15V 0.2A	12.5-16.5V H/L
RM24-005	24V 0.2A	21.0-26.5V H/L

TDK

RMシリーズ ¥50,000 ¥1,500



RM05-005	5V 20A	4.5-5.5V H/L
RM05-005	5V 10A	5.0-10.0V H/L
RM10-005	12V 0.2A	10.0-13.5V H/L
RM15-005	15V 0.2A	12.5-16.5V H/L
RM24-005	24V 0.2A	21.0-26.5V H/L

オフィスコンピュータ・マイクロコンピュータ・電子パーツ
業務無線・システム情報機器・関係書籍



NASAマイコン

NASAコンピュータ事業部 ☎(0552) 53-7373 (代)

甲府市道部一丁目9-10

Byte Shop

LKIT-8 オリジナル ファミリー



Byte Shopオリジナル セット

Aセット ¥95,000

■ Lkit-8 ■ 電源 5V2A

Dセット ¥226,700

■ Lkit-8 ■ 電源 5V10A
■ プリンター TSP-7706B
■ VRAM MB2504
■ 4Kスタティックメモリボード KEMB-001K
■ ラック KERO-008K

Bセット ¥130,000

■ Lkit-8
■ 電源 5V2A
■ プリンター TSP-7706B

Cセット ¥160,000

■ Lkit-8
■ 電源 5V2A
■ プリンター TSP-7706B
■ オーディオ カセット TC1100

他にも
各種ありますので御利用下さい。

★フルキーボード Logites K-12

5V単一電源、エンコーダ内蔵

★P-ROMプログラマ Logites K-13

Lkit-8に接続、P-ROM書き

★ビデオ・コントロール・ユニット Logites K-15

キーボードにビデオインターフェース、電源内蔵

★CRTディスプレイ装置 Logites K-101

9" CRT高解像度、インターフェース、電源内蔵

★カセット式デジタル磁気テープ記憶装置 Logites K-201

キヤクMT-2使用、電源、インターフェース内蔵

★ビデオ・ターミナル Logites 1000

キーボード(K-15)+9" CRTディスプレイ(K-101)

◆限定販売◆

★ミニドットプリンター

Logites K-11 ¥39,500

★4K Static RAMボードキット

KEMB-001K ¥18,000
ボード・周辺パーツ・IC付、RAM(MB102 32個) 別売
ボードはガラスエポキシ、スルーホール、レジスト処理を行なった高信頼性です。

★16K EPROMボード KEB-001

メモリ-MB9518 16個、4Kバイト単位アドレス設定
★ Lkit-8用小型フラッシュキット KERO-008K ¥19,500

★ミニプロセッサ・ディスクドライブ SA-400

(ディスク1枚 ¥2,900) ¥168,000

★ユニバーサル基板 KEUB-001

(H68用 100Pもあり) ¥7,900

パーソナルな利用ができる マイ・コンピュータ

16 Bit Microcomputer Kit LKIT-16

■ 16ビットCPU ■ RAM0.5K語(最大1K語)
■ ROM1K語(最大2K語)/完全キット、詳細マニュアル付

LKIT-16に

高度なインタフェース・モジュールが登場。
さらに自由なシステム構成が展開。

拡張メモリボード (RAM1K語・最大3.25K語/ROM・最大2K語)をはじめ、プリンタ、テレビ、カセット、テレビ型のインタフェースが新登場。テレビ型インタフェースはカラーグラフィックとキャラクターを同時に表示できますので本来に見られなかった高度なテレビゲームや図形処理ができます。さらに、複数台の周辺機器が同時に使用できるようシステム構成されています。

■ Lkit-16 本体	¥98,000
■ 拡張メモリボード	¥42,000
■ マザーボード	¥11,800
■ プリンタインターフェース(完成品)	¥24,800
■ テレビインタフェース	¥39,000
■ テレビインタフェース オプション	¥29,000
■ カセット・テレビ型・インタフェース	¥17,500
■ 電源(本体用)	¥17,000
■ 放電プリンタ EUY-10E	¥16,000
■ 電源 5V10A、+12V1A、5V1A	¥45,000



マイコンは

使いこなす時代。

豊富な周辺機器を使い、多岐的なテレビゲームをはじめ、あなたの身の周りのアイデアに駆動。

福岡 Byte Shop	岡谷 Byte Shop	関東 Byte Shop	大阪 Byte Shop	名古屋 Byte Shop	伊勢崎 Byte Shop
〒812 福岡市南区南門外1-6 電話(092)713,312/296	〒384 長野県岡谷市中央4-11 五斗川ビル 電話(0266)2,318/75	〒112 東京都千代田区有明1-15-16 有明センタービル4F 電話(03)231,5074	〒556 大阪市西区南堀江4-6-5 電話(06)544,114/48	〒460 愛知県名古屋市中区大宮1-15-15 パナソニックビル 電話(052)261,029/30	〒372 群馬県伊勢崎市中央1-15-5 電話(0270)3,356/2

MICROPROCESSOR and SUPPORT CHIP

AM9080 ADC 特価 ¥2,800	DM18L551 8bit 75V 5V5L 382
AD7520 0.05% 10bit 1,000	DM18L551 8bit バッファ ¥382
AD7520 0.05% 10bit 1,000	DM18L551 8bit バッファ ¥382
AD7520 0.05% 10bit 1,000	DM18L551 8bit バッファ ¥382
AD7520 0.05% 10bit 1,000	DM18L551 8bit バッファ ¥382
AD7520 0.05% 10bit 1,000	DM18L551 8bit バッファ ¥382
AD7520 0.05% 10bit 1,000	DM18L551 8bit バッファ ¥382
AD7520 0.05% 10bit 1,000	DM18L551 8bit バッファ ¥382
AD7520 0.05% 10bit 1,000	DM18L551 8bit バッファ ¥382
AD7520 0.05% 10bit 1,000	DM18L551 8bit バッファ ¥382

AD0 ELECTRONIC INDUSTRIAL CO. LTD. 10-7 Sotokanda 3 Chiyoda TOKYO

SN74シリーズ () の価格をLS

7400		7401	7402	7403	7404	7405	7406	7407	7408	7409	7410	7411	7412	7413	7414	7415	7416	7417	7418	7419	7420	7421	7422	7423	7424	7425	7426	7427	7428	7429	7430	7431	7432	7433	7434	7435	7436	7437	7438	7439	7440	7441	7442	7443	7444	7445	7446	7447	7448	7449	7450	7451	7452	7453	7454	7455	7456	7457	7458	7459	7460	7461	7462	7463	7464	7465	7466	7467	7468	7469	7470	7471	7472	7473	7474	7475	7476	7477	7478	7479	7480	7481	7482	7483	7484	7485	7486	7487	7488	7489	7490	7491	7492	7493	7494	7495	7496	7497	7498	7499	7500	7501	7502	7503	7504	7505	7506	7507	7508	7509	7510	7511	7512	7513	7514	7515	7516	7517	7518	7519	7520	7521	7522	7523	7524	7525	7526	7527	7528	7529	7530	7531	7532	7533	7534	7535	7536	7537	7538	7539	7540	7541	7542	7543	7544	7545	7546	7547	7548	7549	7550	7551	7552	7553	7554	7555	7556	7557	7558	7559	7560	7561	7562	7563	7564	7565	7566	7567	7568	7569	7570	7571	7572	7573	7574	7575	7576	7577	7578	7579	7580	7581	7582	7583	7584	7585	7586	7587	7588	7589	7590	7591	7592	7593	7594	7595	7596	7597	7598	7599	7600	7601	7602	7603	7604	7605	7606	7607	7608	7609	7610	7611	7612	7613	7614	7615	7616	7617	7618	7619	7620	7621	7622	7623	7624	7625	7626	7627	7628	7629	7630	7631	7632	7633	7634	7635	7636	7637	7638	7639	7640	7641	7642	7643	7644	7645	7646	7647	7648	7649	7650	7651	7652	7653	7654	7655	7656	7657	7658	7659	7660	7661	7662	7663	7664	7665	7666	7667	7668	7669	7670	7671	7672	7673	7674	7675	7676	7677	7678	7679	7680	7681	7682	7683	7684	7685	7686	7687	7688	7689	7690	7691	7692	7693	7694	7695	7696	7697	7698	7699	7700	7701	7702	7703	7704	7705	7706	7707	7708	7709	7710	7711	7712	7713	7714	7715	7716	7717	7718	7719	7720	7721	7722	7723	7724	7725	7726	7727	7728	7729	7730	7731	7732	7733	7734	7735	7736	7737	7738	7739	7740	7741	7742	7743	7744	7745	7746	7747	7748	7749	7750	7751	7752	7753	7754	7755	7756	7757	7758	7759	7760	7761	7762	7763	7764	7765	7766	7767	7768	7769	7770	7771	7772	7773	7774	7775	7776	7777	7778	7779	7780	7781	7782	7783	7784	7785	7786	7787	7788	7789	7790	7791	7792	7793	7794	7795	7796	7797	7798	7799	7800	7801	7802	7803	7804	7805	7806	7807	7808	7809	7810	7811	7812	7813	7814	7815	7816	7817	7818	7819	7820	7821	7822	7823	7824	7825	7826	7827	7828	7829	7830	7831	7832	7833	7834	7835	7836	7837	7838	7839	7840	7841	7842	7843	7844	7845	7846	7847	7848	7849	7850	7851	7852	7853	7854	7855	7856	7857	7858	7859	7860	7861	7862	7863	7864	7865	7866	7867	7868	7869	7870	7871	7872	7873	7874	7875	7876	7877	7878	7879	7880	7881	7882	7883	7884	7885	7886	7887	7888	7889	7890	7891	7892	7893	7894	7895	7896	7897	7898	7899	7900	7901	7902	7903	7904	7905	7906	7907	7908	7909	7910	7911	7912	7913	7914	7915	7916	7917	7918	7919	7920	7921	7922	7923	7924	7925	7926	7927	7928	7929	7930	7931	7932	7933	7934	7935	7936	7937	7938	7939	7940	7941	7942	7943	7944	7945	7946	7947	7948	7949	7950	7951	7952	7953	7954	7955	7956	7957	7958	7959	7960	7961	7962	7963	7964	7965	7966	7967	7968	7969	7970	7971	7972	7973	7974	7975	7976	7977	7978	7979	7980	7981	7982	7983	7984	7985	7986	7987	7988	7989	7990	7991	7992	7993	7994	7995	7996	7997	7998	7999	8000	8001	8002	8003	8004	8005	8006	8007	8008	8009	8010	8011	8012	8013	8014	8015	8016	8017	8018	8019	8020	8021	8022	8023	8024	8025	8026	8027	8028	8029	8030	8031	8032	8033	8034	8035	8036	8037	8038	8039	8040	8041	8042	8043	8044	8045	8046	8047	8048	8049	8050	8051	8052	8053	8054	8055	8056	8057	8058	8059	8060	8061	8062	8063	8064	8065	8066	8067	8068	8069	8070	8071	8072	8073	8074	8075	8076	8077	8078	8079	8080	8081	8082	8083	8084	8085	8086	8087	8088	8089	8090	8091	8092	8093	8094	8095	8096	8097	8098	8099	8100	8101	8102	8103	8104	8105	8106	8107	8108	8109	8110	8111	8112	8113	8114	8115	8116	8117	8118	8119	8120	8121	8122	8123	8124	8125	8126	8127	8128	8129	8130	8131	8132	8133	8134	8135	8136	8137	8138	8139	8140	8141	8142	8143	8144	8145	8146	8147	8148	8149	8150	8151	8152	8153	8154	8155	8156	8157	8158	8159	8160	8161	8162	8163	8164	8165	8166	8167	8168	8169	8170	8171	8172	8173	8174	8175	8176	8177	8178	8179	8180	8181	8182	8183	8184	8185	8186	8187	8188	8189	8190	8191	8192	8193	8194	8195	8196	8197	8198	8199	8200	8201	8202	8203	8204	8205	8206	8207	8208	8209	8210	8211	8212	8213	8214	8215	8216	8217	8218	8219	8220	8221	8222	8223	8224	8225	8226	8227	8228	8229	8230	8231	8232	8233	8234	8235	8236	8237	8238	8239	8240	8241	8242	8243	8244	8245	8246	8247	8248	8249	8250	8251	8252	8253	8254	8255	8256	8257	8258	8259	8260	8261	8262	8263	8264	8265	8266	8267	8268	8269	8270	8271	8272	8273	8274	8275	8276	8277	8278	8279	8280	8281	8282	8283	8284	8285	8286	8287	8288	8289	8290	8291	8292	8293	8294	8295	8296	8297	8298	8299	8300	8301	8302	8303	8304	8305	8306	8307	8308	8309	8310	8311	8312	8313	8314	8315	8316	8317	8318	8319	8320	8321	8322	8323	8324	8325	8326	8327	8328	8329	8330	8331	8332	8333	8334	8335	8336	8337	8338	8339	8340	8341	8342	8343	8344	8345	8346	8347	8348	8349	8350	8351	8352	8353	8354	8355	8356	8357	8358	8359	8360	8361	8362	8363	8364	8365	8366	8367	8368	8369	8370	8371	8372	8373	8374	8375	8376	8377	8378	8379	8380	8381	8382	8383	8384	8385	8386	8387	8388	8389	8390	8391	8392	8393	8394	8395	8396	8397	8398	8399	8400	8401	8402	8403	8404	8405	8406	8407	8408	8409	8410	8411	8412	8413	8414	8415	8416	8417	8418	8419	8420	8421	8422	8423	8424	8425	8426	8427	8428	8429	8430	8431	8432	8433	8434	8435	8436	8437	8438	8439	8440	8441	8442	8443	8444	8445	8446	8447	8448	8449	8450	8451	8452	8453	8454	8455	8456	8457	8458	8459	8460	8461	8462	8463	8464	8465	8466	8467	8468	8469	8470	8471	8472	8473	8474	8475	8476	8477	8478	8479	8480	8481	8482	8483	8484	8485	8486	8487	8488	8489	8490	8491	8492	8493	8494	8495	8496	8497	8498	8499	8500	8501	8502	8503	8504	8505	8506	8507	8508	8509	8510	8511	8512	8513	8514	8515	8516	8517	8518	8519	8520	8521	8522	8523	8524	8525	8526	8527	8528	8529	8530	8531	8532	8533	8534	8535	8536	8537	8538	8539	8540	8541	8542	8543	8544	8545	8546	8547	8548	8549	8550	8551	8552	8553	8554	8555	8556	8557	8558	8559	8560	8561	8562	8563	8564	8565	8566	8567	8568	8569	8570	8571	8572	8573	8574	8575	8576	8577	8578	8579	8580	8581	8582	8583	8584	8585	8586	8587	8588	8589	8590	8591	8592	8593	8594	8595	8596	8597	8598	8599	8600	8601	8602	8603	8604	8605	8606	8607	8608	8609	8610	8611	8612	8613	8614	8615	8616	8617	8618	8619	8620	8621	8622	8623	8624	8625	8626	8627	8628	8629	8630	8631	8632	8633	8634	8635	8636	8637	8638	8639	8640	8641	8642	8643	8644	8645	8646	8647	8648	8649	8650	8651	8652	8653	8654	8655	8656	8657	8658	8659	8660	8661	8662	8663	8664	8665	8666	8667	8668	8669	8670	8671	8672	8673	8674	8675	8676	8677	8678	8679	8680	8681	8682	8683	8684	8685	8686	8687	8688	8689	8690	8691	8692	86
------	--	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	----

ロビン電子産業

TVゲームから無限の応用を求める。

COMKIT 8061



BASIC COMPUTER KIT

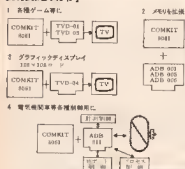
特徴

COMKIT 8061は4K BASICを使用し低コストかつ高信頼性のあるシステム構成で高級言語BASICをプログラミングできます。オプションの追加によりグラフィックディスプレイ、24チャンネルのリークのオンオフ制御等、教育用としてだけでなく自動制御、計測制御等の工学分野への応用として最適です。

仕様

- 1 CPU: SC MP-II 4MHzクロック
- 2 BASICROM: 4K BASIC (1K PROM x 4)
- 3 RAM: M-2114最大16個(8Kバイト)実装は1Kバイト
- 4 入力: 61キーボード ASCII 配列、英数字 特殊文字・カナ文字使用可
- 5 出力: 家庭用TV、VHF 20ch出力
- 6 外部記憶: オードイオカセット機構装備
- 7 拡張性: バッファ付拡張端子あり、DMAコントロール
- 8 電源: AC 100V 15W 5V3A ±12V 0.2A
- 9 外形寸法: 340 x 320 x 105

拡張応用例



拡張用製品

マザーボード

★ADB-003 TVD-02スロット付 全4スロット ★ADB-004 拡張用マザーボード 4スロット 4スロット単位で拡張可能 各種ディスプレイユニット

★TVD 01 64x32ドット白黒ドットディスプレイDMA方式ゲーム用(マザーボード使用不可) ★TVD-03 64x32ドット(32x32ドット)カラーディスプレイDMA方式ゲーム用(マザーボード使用不可) ★TVD-04 128x104ドット点画(3色からカラー可能)グラフィックディスプレイ ★TVD-05 64x16ドットカラーディスプレイ簡易グラフィック用メモリーボード

★ADB-001 12K ROM、RAMポート ROM2708 4個 RAM2114 16個 ★ADB-005 4KバイトRAMポート2702 32個 ★ADB-006 16KバイトRAMポート 2114 32個 ★ADB-008 2708 2756 2716用PROMボード

1. Oボード

★ADB-011 24チャンネル入力 24チャンネルフェットカプ出力 ★ADB-009 フロッピーディスクコントローラ(開発中)

(注) TVD-01、03以外はすべてマザーボード使用可

EXP-344(拡張セット)

ADB-004マザーボード 1枚
44Pコネクタ(マザーボード用) 4本
44Pコネクタ(インデックタイプ) 2本
34Pプラグケーブル 1組
ビニール平封コード

5月中旬発売予定

デモマシン作動中!

5月8日~20日マイコンコンピュータショーにてデモンストレーションを行ないます 乞願期待!

(予約発売致します)

COMKIT 8060

世界一安い BASIC COMPUTER //



超低価格 BASIC COMPUTER KIT

特徴

COMKIT 8060は4K BASICを使用し低コストかつ最も単純なシステム構成で高級言語BASICをプログラミングできます。学校・会社等の教育用として又アマチュアの遊び用として最適なコンピュータです。

仕様

- 1 CPU: SC MP-II 4MHzクロック
- 2 BASICROM: 4K BASIC (2K PROM x 2)
- 3 RAM: M-2114 最大12Kバイト迄拡張可(実装1Kバイト)
- 4 入力: 61キーボード ASCII 配列、英数字 特殊文字使用可
- 5 出力: VHF 20ch出力(家庭用TVのアンテナ端子に接続するだけ)画面サイズ 32cm x 16cm
- 6 外部記憶: オードイオカセット (インターフェースは別売り)
- 7 電源: AC 100V 15W 5V3A ±12V 0.2A
- 8 外形寸法: 340 x 320 x 105

ホビーショップ(マイコンコーナー)オープン

新潟: 新潟県新潟市万代1-5-1 スーパーダイエーパーツコーナー
神奈川: 神奈川県横浜市鶴岡2030 東急ハンズホビーコーナー
東京: 東京都千代田区神田佐久間町1-14 第二東ビル306号室

0252 41 5111 月367 担当: 目黒
0456-26-8735 担当: 梅津
03 255-6027 担当: 中島

技術相談受付中: アドテック、日立、東芝、パナファコム、松下、他

※日・時・問は各ホビーショップへお聞き下さい。

●ご注文は現金書留・郵便にて。住所・氏名・品名・個数・郵便番号をしっかりと書いてお願致します。

●送料: 5,000円以下→〒200/5,000円以上→〒300

●多数お買い上げの方には、別途見舞い致します。地方県営、ユーザー、メーカー大歓迎!

CR、RTL、CMOS、マイコン、総合商社の
ロビン電子産業(株) I/O係

〒101 東京都千代田区神田佐久間町1-14 第二東ビル306号室
03-255-6027 営業時間: 9:30~19:00 休日: 日曜日・祭日

●当店はビル3階のため来店の際は階2にお乗り下さい(10階)と書いて下さい。(車口及び地下鉄の方、駅より50mです。)





エンタープライズからの 贈り物?

BASIC COMPUTER KIT!
¥99,800(千円税別)

COMKIT8060!

学習用、教材用キット

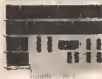
家庭用 TV を用意するのみに、すぐ高級言語 BASIC が使用出来ます。豊富なマニュアル類で理解も早く、学校教材用、アマチュア用、会社社員教育用としても最適です。工業用 BASIC を使用しているため各種制御のシミュレーション等にも利用できます。

- ①NIBL BASIC 4KROM内蔵
- ②RAM 1K内蔵 12Kバイト拡張可
- ③61フルキーボード付
- ④TVインターフェース付
- ⑤電源、ケース付
(CPUボード、キーボード、電源はキットです)
- ⑥マニュアル
NIBL BASIC 文法説明書、組立説明書
BASIC入門書、アプリケーションプロ
グラム25例。

オプション

- ・カセットインターフェース ¥7,000
- ・専用ファン ¥8,000

ADB-006 / 16K RAMボード



16K RAMボードADB-006 (8080 / 6800両用)

ADB-001、003~5共通仕様で拡張が容易、MB-01、02マザーボード使用可。

- ADB-006A 4K RAM付 ¥39,500(千500)
- ADB-006B 8K RAM付 ¥66,500(千600)
- ADB-006C 16K RAM付 ¥92,500(千900)

TVD-04 ¥34,500(千500)

- 128×128ドット カラーグラフィックディスプレイ
- TVD-04 3ユニットで3色のカラーグラフィック、又は白黒の場合は8段階の濃度をだすことが可能。
- TVD-04 1ユニットなら白黒のグラフィック。

- ドットサイズ 128(横)×128(縦)
- RAM 4Mbit
- ワンキーボードあり
- 2チャンネルカラーの場合はドットグラフィック(カラー方式)
- 24ピンコネクタ 4ピンマザーボードに接続可能(128K RAM RAMボード 4ピンメモリボードと同一マザーボードで利用可)
- 外形サイズ 170×120(TVD-02と同一サイズ)
- 電源 5V 1A
- 14ピンメモリ 2Kビット

※注文は現金書留、振替(振込431)、振替又は銀行送金(第一勧業銀行西口支店、当座0109194)でお願いします。
尚歩額(2,000円以下)は切手にても可(但し100円以下の切手)、休日、日曜、祭日、おし月の第一日曜日は営業致しません。

株式会社 アドテック システム サイエンス
〒220 横浜市区西平沼2-3-17 TEL 045(324)1290

★サウスウェスト社全製品取扱中・カタログを御請求下さい



新時代のホームコンピュータM200シリーズの下位機種

M100シリーズ新たに登場

白黒TV+オーディオカセット装置付 フルオプション¥299,000
本体¥199,000

OPTION
モニター



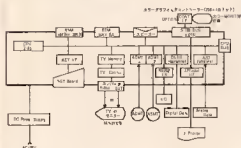
M100本体

M100の概要

M100は簡易な8KBASIC言語を使用し、S100バスを装備した拡張性に富むシステムです。他のホームコンピュータとの通信、電話による通信はもとより、ちょっとした身のまわりの電器製品、機械の制御、日常的なメモの記憶、その印字出力、高級ゲームにご使用可能です。又、64桁、24行の表示面をもつモニター、テレビと対話しながら16個のオン・オフ接点を制御し、まわりの温度、湿度、圧力の検出装置からのデータを取り込み、計算します。メモリは標準で16Kバイトを内装し、将来48Kバイトまで拡張でき、ROMも8Kバイトまで御使用頂けます。システムはBASICプログラムにより操作され、BASIC命令キーが約20個用意されております。

M100の機能

- 標準構成 ●CPUにZ-80使用 ●メモリ16KバイトからMAX32Kバイト(内蔵可) ●64×24文字(白黒)モニター、テレビ・コントロール内蔵 ●オーディオカセット2台接続可能 ●アナログ信号変換器2台内蔵(8bit ADG)
- デジタル出力各8本内蔵(リレードライバ、ホットカプラ付)
- ジョイスティックレバー1本内蔵 ●2オクターブ音域のスピーカ内蔵 ●フルキーボード及びベーシックキーボード ●ジャーナルプリンターインターフェース内蔵 ●S100バス使用可能
- オプション ●5V4A電源 ●白黒モニターTV、カラーモニターTV ●カラーグラフィックコントローラ(M100、M200共用可)
- オーディオカセット装置 ●S100バス用拡張筐体 ●AC100V電源コンセントコントローラ



M200シリーズ 好評発売中!! ソフトも豊富になりました。

- ガソリンスタンド
- タクシー会社用
- 会計事務所
- 歯科医
- 在庫管理

¥1,390,000

デモンストレーション中!!



●ガンシンショッピングローンが使えます。

お支払い方法(ローン、リース、買取とご自由にお選び下さい。)

《MODEL CT-1 只今実演中》

CT-1 コンピュータ(スピーチシミュレーター)	¥ 230,000
TK-80	¥ 88,500
TK-80E	¥ 67,000
TK-80BS	¥ 128,000
PROMライターモジュール (2708、2716、8518用)	¥ 32,000

PROMイレーサー(タイマー付)	¥ 20,000
シリアルインタフェースモジュール	¥ 44,700
東芝 8ビットマイクロコンピュータキット(TVインタフェース、CMTインタフェース付)	¥ 85,000
NECフルキーボード KBR015-002	¥ 52,000
NECフルキーボード KBR014-002	¥ 49,800
	各¥1,000

●マイコンのカタログ請求は社名と機種名を指定して〒200を添えてお申し込み下さい。

サンシンショップ

(株)三真電機

SHOP

〒101 東京都千代田区外神田1-10-11
東京ラジオアパルト地下 ☎03(253)6686代
本社
〒101 東京都千代田区外神田3-2-16
加藤ビル3階 ☎03(253)2621代

思いのままに使いこなそう!!

貴方も自分専用のマイコンをもって自由自在に使ってみませんか?

(ローン取扱い致します。
御相談下さい。(3~36回))

331VCRT完成品 ¥43,000 5V、0.5A、

MT-2 デモ中 / H-68 ¥99,700

メモリー
4Kボード
のみ ¥12,000
¥9,000

プロセシー

6800

CT64 ¥185,000

¥480,000

¥179,500

AC-30 カセット
¥45,000

CMTソフト箱 (6800, 8080)

NEC TK-80

BS

ソフトベシツク

S100BUS

インターフェイス・シヨツプ

大阪 ICM

〒555 大阪市東淀川区東中島5-1-15 伊藤商会ビル5F
06(44)1131 営業部06(44)0079(代) 02本館日本橋

¥43,000

ディスプレイロー ¥250,000

バグシステム ¥121,800

Do it yourself!! 手作りマイコンEX-80には、 マイコンキットとして望まれた機能の 数々が装備されています。



¥85,000 (送料¥1,000)

TLCS-80A EX-80(組立キット)は、8ビットのCPU(TMP9080AC)を中心にしてMOS LSIと各種部品で構成された完全部品キットの1作
りマイコンです。マイコンそのものとしての回路はもちろん、テレビ
インタフェース回路やオーディオカセット・インタフェース
回路がワンボード上に搭載されています。また、これらの
機能が組み合わされ、即座動できるROM(TMM331AP)に
モニタープログラムが格納されています。さらに、16進キー
ボード使用の際、テレビの音声回路を利用してキー入力の確
認もできます。EX-80は、マイコンキットとして望まれた機能の数々
を装備して操作性と応用能力を高めた「システム・マイコン」です。

■主な仕様 ●CPU TMP9080AC(8ビット処理) ●クロックジェネレータ T101822
4P ●システム・コントローラ T1018228P ●ROM TMM331AP(2Kバイト)のモニタープログラム
A ●RAM TMM314P(最大2Kバイトまで増設可能) ●PROM:オプション ●TMM322C ●キーボ
ード:25KBY, DATA(16KBY), FUNCTION(9KBY) ●LED-DISPLAY:セグメントLEDによる16進表示
(ADDRS:84bit, DATA:4bit) ●キャラクタジェネレータ T3477A ●テレビンクジェネレータ TC5003P(同調
信号発生器) ●プログラマブル・コミュニケーション・インタフェース TMP9551C ●その他:EX-80基板、各種IC etc.

NEC μCOM Basic Station TK-80BS

価格 ¥128,000
(送料¥1,300)
電源別売



■TK-80BSの性能 ●構成:キーボード、ペーパーバックステーションボード、マザーボ
ード、100ピンコネクタ、ユーザーズマニュアル(電源別売) ●バス:TK-80バス、マザー
ボードが拡張性があります ●メモリー:RAM7K(4bit), RAM12K(4bit)(ソケット)
付き、標準搭載 RAM5K(4bit), ROM8K(4bit) ●モニタ:RAM2K(4bit), BASIC4K(4bit)
ビデオディスプレイ:32文字×16行、文字とグラフィック記号が同時に使えます ●
文字の種類:JIS標準漢字、数字、記号、カナ、特殊記号、128種/ASCII標準
漢字、数字、記号、64種/小文字の半角、小文字のギリシヤ文字、記号、64種
グラフィック記号、128種/半角・25種 ●画面コントロール:画面内容はプログラ
ムで可、ファームウェアによる自動スクローリング ●キーボード:JIS標準VX100の55キ
ー、特殊キー、100個の数字キー、英記号、カナ記号キーボードにソフトキー機能
で切り換えるので操作がやり易くなります ●カセットテープインターフェイス:記
録方式、カセットタイスタター(300回転)テープスピードの電動に準じ、信頼性の
ある回路を採用しています ●シリアルインターフェイス:75/110/150/300/600/1200
/2400bpsにて切り換えて使用できます。併し、シリアルインターフェイスはカセットイン
ターフェイスと併用します ●モニタープログラム:メモリー内容の表示/電圧、レジスタ
内容の表示/電圧、メモリープログラム、1命令毎の電圧と実行開始アドレス、ブレイ
クポイントの設定、カセットテープの書き込み/読み出し(ベキサマワード)、
カセットテープの書き込みデータとの比較チェック機能 ●BASIC:配列と機械的
プログラムとの組み合わせに便利なCALL、PEEK、POKE機能 ●ソフトウェア(開
発中のものを含む):5Kバージョン、8Kバージョン・セルフチェックプログラム/エディター、そ
の他各種応用プログラム ●消費電源:+5V3.5A以下(RAMフル実装)、+12V
0.3A以下(TK-80を含む機です)

μCOM Training Kit TK-80E ¥67,000
(送料¥1,000)

各社マイクロコンピュータ

日立H8B/TR ¥99,500 (¥1000)	モトローラMEK68000IB
オプティムH472114	(SPEED MASTER電源内蔵)
専用電源:HTP503¥11,500(¥1000)	(完成品) ¥93,000(¥1000)
H68/TR:ミニマル ¥2,000(¥350)	東芝TLCS12AEX ¥57,000(¥1000)
ファコムLK1T-B ¥85,000 (¥1500)	NEC TK-80 ¥84,500(¥1000)
パナソニックKIT-16 ¥98,000	キヤノンTPD5101C ¥2,000
(¥1000) オプティムMH8111 ¥710	専用電源:IR ¥5,800(¥1000)
専用電源 ¥17,000(¥1000)	インテルSDK-85 ¥81,000(¥1000)

TK-80部品の周辺機器

- カセット・インターフェース
IC-0008 専用FS3300ビット電源5V ¥8,500
- カセット・ビデオ・ディスク駆動装置
MT-2 ディスク TK-80機対応付 電源5V, 11V ¥95,000 (オプション ¥2,000)
- 紙テープリーダー-HR-100(リコー電子)
マイコン入出力用 T11レベル並列出力、電源5V(TK-80機対応付) ¥18,800
- JIS配列標準キーボード
KBRO14-001 ・テンキーなし、エンコード付 ¥55,000
KBRO15-001 ・テンキー付、エンコード付 ¥61,500
- 黒面ディスプレイ・デジタルTV-32A 32×32ドット、電源5V ¥29,500
- カラーディスプレイ・モジュールTV-64C
64×64ドット、4色・8ビットRAM方式、1024バイト電源5V ¥44,000
- キャラクタディスプレイ・モジュール
TV-CD 32×24文字×16行、ライトペン機能付、電源5V, 12V ¥46,000
- 4KRAM拡張ボード μPD2102A, 32ビット 電源5V ¥18,000
- 4KROM拡張ボード μPD4540, 16ビット 電源5V ¥18,000
- 増設用マザーボード3 ¥5,500
- バックアップボード ¥4,300
- 電源 TK-80専用IR-15 ¥5V3.5A以下 IC-0004 ¥10,500
- TK-80BS専用電源C0005 ¥5V3.5A/12V0.5A, 5V0.1A/12V2.0A(¥1,000)

その他の周辺機器

- H8B/TR用TVインターフェース・モジュールH8BTV01 ¥69,500
- 放電プリンタ TSP-7706B ・キャビネ内蔵 直流マイコンのPIAに準
拠、電源付(TK-80, MEK-6800, H68/TR, LK1T6等) ¥37,000
- TKマイコン用電源
TRMO23 +5V(5A)・+12V(0.3A) -5V(0.3A/100mA)S-電源 ¥36,000
RM05-03 +5V(5A)・4.5V-5.5V可変 ¥25,000
- シンケンマイコン用電源ICAS-3500 +5V(3A) ¥13,000
- LK1T16用ビデオ・キャラクター・ディスプレイ
LA05K-A 640文字(16行×40列) ¥38,000
- LK1T16用カラービデオ・キャラクター・ディスプレイ
LA05K-A1 カラー画面(160×80ドット) ¥29,000
- LK1T-16用拡張メモリアボードLA02K-A ¥42,800
- LK1T-16用プリンタ・インターフェースLA05K-B ¥24,800
- LK1T-16用キーボードLA15K-A ¥11,800
- LK1T-16用カセット・ディスプレイ・インターフェースLA05K-D ¥17,500

(営) ●品目: 各社マイコン・半導体全製品・性能器・プリント基板・電子部品一式

田中無線

〒100 東京都千代田区外神田3-13-3本舗 ¥255-5151(代)
マイコン半導体部 ¥253-3201





MEM⇄COPY, MASTER→COPY他
多機能モード!

2708, 2704
8080系P-ROMライター・モジュール
MULTI-87A ¥32,000



<仕様>

- 動作モード——MEM⇄COPY, MASTER→COPY, MASTER→MEM, 消去チェック
- 制御——N-88バス, 8080にてプログラム制御
- P-ROM接続——オンボードまたはフラットケーブルにて接続
- ソフトウェア——リストにて提供, 256バイト
- サイズ——130×165mm

P-ROM イレーサー
NPE-41 ¥20,000



<仕様>

- 消去時間——連続または0-30分タイマー
- 紫外線長さ——2537オングストローム
- 消去回数——最大6回
- 点灯方式——グロースタート式
- 電源周波数——50Hz用と60Hz用あり
- サイズ——80H×140W×170Dmm

製造元

ノーゼル・エンジニアリング(株)

〒168 東京都杉並区宮前4-24-21

TEL03(331)1571

NOZEL

- 米国のセンスサイザー専門誌"SYNAPSE"好評入荷中(バックナンバー有り) 各¥600 (〒120)
- カタログ請求は型番明記の上, 切手100円分を同封してハックス・エレクトロニカ営業部1/0係まで申し込み下さい。
- ノーゼル・ブランドの製品についての技術的なお問い合わせは, ノーゼル・エンジニアリング技術部 野崎までお願いします。
- 大学研究室には指定の様式で納入いたします。

PIAによるクイック・インターフェイス!

7201, 2718
6800系P-ROMライター・モジュール
PPW-01 ¥32,000



<仕様>

- 動作モード——MEM⇄COPY
- 制御——4 $\frac{1}{2}$ ピンチ44P(ユーザー・プログラマブル)6800にてプログラム制御
- P-ROM接続——ゼロブレッシャー・ソケット
- ソフトウェア——MEK-DII用カセット及びリスト付, 160バイト
- サイズ——115×155mm (KELサイズ)

4Kスタチックメモリボードキット
PM-02 (4K実装) ¥36,000



<仕様>

- 16ビット・フルアドレスデコード
- 1Kバイト単位メモリ・プロテクト
- 単 5V電源(電流1A Max.)
- アドレス, データ・バッファ付(8126相当)
- アクセスタイム500ns Max.
- コネクタ 44P KEL22×2Pサイズ

PM-02N (メモリなし基板) ¥18,000

ノーゼル・エンジニアリング販売代理店

ハックス・エレクトロニカ株式会社

営業部/〒150 東京都渋谷区宇田川町 2-1
渋谷ホームズ810 TEL03(496)2946

★官公庁・学校関係は所定の様式及支払手続で全品種の注文をお受けします。

代引取扱★各社半導体全品種取扱★一級新品

モトローラ電源レギュレーター

MC7805 5V最大1.5A ¥300

MC7812 12V最大1.5A ¥300

MC7815 15V最大1.5A ¥300

μPDI4308(8V1ANE) ¥180

MC78M05 15各50ヶ以上 @ ¥210

専用フィン付6枚(25×15mm)黒メッキ ¥100

2SB554/2SD424

無至実効出力 160W

コンプリ大特売 ¥1,980

V_{CE} 180V V_{CE} 180V Ic 15A PC150W

To-3 東芝 電力増幅・HiFi

●hFEバランス±5%内特選品

2SC2092

高周波電力 ¥180

V_{CE} 120V V_{CE} 120V Ic 15A PC150W

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

2SB555 2SD425

¥1,480

大特別選別

V_{CE} 140V V_{CE} 140V

Ic 12A P_C 100W

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

●hFEバランス±5%内特選品

2SA627/2SD188

特選コンプリ

hFEバランス±5%内

1組 ¥980

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

hFEバランス±5%内

日立 モスFET

2SK134/2SJ49

¥3,400

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SJ50/2SK135

¥4,900

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

P_C 100W To-3

2SD316 2SD287

¥380

V_{GS} 120V I_{SS} 7A

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

Ic 7A P_C 50W

To-3型(東芝) 永年保証付

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

Ic 7A P_C 50W

To-3型(東芝) 永年保証付

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

Ic 7A P_C 50W

To-3型(東芝) 永年保証付

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

Ic 7A P_C 50W

To-3型(東芝) 永年保証付

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

Ic 7A P_C 50W

To-3型(東芝) 永年保証付

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

Ic 7A P_C 50W

To-3型(東芝) 永年保証付

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

Ic 7A P_C 50W

To-3型(東芝) 永年保証付

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

Ic 7A P_C 50W

To-3型(東芝) 永年保証付

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

Ic 7A P_C 50W

To-3型(東芝) 永年保証付

2SC1172B

¥1,150

V_{CE} 150V V_{CE} 150V

2SD710(赤凸形) ¥170

SLP711(赤凸形) ¥170

金銀ブラット付 発光ダイオード

10ヶ ¥1,500

10ヶ以上 ¥1,100

10ヶ以上 ¥1,100

MC8T26 ¥700	モトローラ MC8T28 ¥700
MC6800L (MPU) (6800Pは ¥6,000)...	¥6,800
MC6802P (6800+6810+6871内蔵).....	¥8,800
MCM6810AP (128×8RAMプラスチック)	¥1,380
MC6820L (PIA) (MC6820Pは ¥3,300)...	¥4,200
MCM6830P-8 (ミグバグ付).....	¥4,800
MC6840P (PTM).....	¥7,100
MC6850L (ACIA) (6850Pは ¥3,300)...	¥4,200
MC6871A (Bは ¥6,700) クロックゼネレータ	¥6,500
MC6573AP (キャラクタージェネレータ)	¥4,800
MCM68708L / 2708L (P.ROM).....	¥8,000

●技術資料● 以上各店 5%引

M-6800 MPUアプリケーション・マニュアル	¥6,000 ¥500
M-6800 MPUプログラミング・リファレンスマニュアル	¥1,500 ¥300
M-6800 MPU 和文マニュアル改訂版	¥2,500 ¥300
マイコンデジター	¥800 ¥300
システムデザインデータブック	
モトローラ C.MOSデータブック	¥1,000 ¥300
モトローラ リニヤICデータブック	¥1,500 ¥300

シャープ大型LED 3Rはアノード 8Rはカソード

GL-9R04-R04 21mm×19mm各 ¥300

9R06-R06 25mm×19mm各 ¥380


9R10-R10 33mm×22mm各 ¥550

8P04 (リフレクト) 21mm×16mm ¥200

YHP型小型LED 7セグメント(カソード) 2桁半 7%×15% ¥180

3桁 7%×15% ¥180

9R06 100ヶ ¥26,000



TLR312	MAN72	シャープ2桁LED ¥480
16ヶ ¥250 10ヶ ¥2,300 100ヶ ¥20,000 意定色中文字 LED カソード	10ヶ ¥180 10ヶ ¥1,650 100ヶ ¥12,000 意定色中文字赤 アノード高19×19mm	GL-6R201 カソード中文字 GL-7R2017 アノード中文字 高18mm×24mm (文字の寸法 9R04と同) 50ヶ ¥19,000 アノード ¥34,000

アンテナックス半円コデ(コテ先別売)	
★半円コデ本体 (S型・A型・B型)	
G型 (15W) TR+IC用	¥2,250 ¥200
G型 (15W) 一電電子用	¥2,800 ¥200
K型 (17W) C.MOSIC用	¥2,480 ¥200
X25型 (25W) 一電電子用	¥2,900 ¥200

★コテ先価格表	
☆C型用	
No.2 (2.8mm) No.1 (1mm)	¥430
No.3 (4mm) No.4 (4.7mm) No.5 (4.7mm)	¥480
No.10 (0.5mm)	¥330
No.102 (2.3mm) No.106 (1mm)	¥620
No.103 (4mm) No.104 (4.7mm)	¥560
No.302 (2.3mm)	¥500

☆G型用 (全部鉄メッキ)	
No.2 (2.3mm) No.21 (3mm) No.212 (4.7mm)	¥580
☆C X型用	
No.1mm-No.8mm (ニッケルメッキ)	¥740
T 2.3mm-8mm-4.7mm-5mm (金メッキ)	¥480
☆X25型用	
2.4mm-8.3mm-4.7mm (金メッキ)	¥640

★交換用ヒーター	
C型用 ¥1,400 C X型用 ¥1,640	
G型用 ¥1,700 X25型用 ¥1,640	
コテスタンド (各型共通) ¥1,380 ¥300	

ご注文は現金書留又は為替で住所氏名・品名をはっきり書いて下さい。

送料 全品に限り合計231円以下 ¥140
送料 ¥300以上 ¥180
送料 ¥500以上 ¥220
送料 ¥1,000以上 ¥260
送料 ¥1,500以上 ¥300
送料 ¥2,000以上 ¥340
送料 ¥2,500以上 ¥380
送料 ¥3,000以上 ¥420
送料 ¥3,500以上 ¥460
送料 ¥4,000以上 ¥500
送料 ¥4,500以上 ¥540
送料 ¥5,000以上 ¥580
送料 ¥5,500以上 ¥620
送料 ¥6,000以上 ¥660
送料 ¥6,500以上 ¥700
送料 ¥7,000以上 ¥740
送料 ¥7,500以上 ¥780
送料 ¥8,000以上 ¥820
送料 ¥8,500以上 ¥860
送料 ¥9,000以上 ¥900
送料 ¥9,500以上 ¥940
送料 ¥10,000以上 ¥980
送料 ¥10,500以上 ¥1,020
送料 ¥11,000以上 ¥1,060
送料 ¥11,500以上 ¥1,100
送料 ¥12,000以上 ¥1,140
送料 ¥12,500以上 ¥1,180
送料 ¥13,000以上 ¥1,220
送料 ¥13,500以上 ¥1,260
送料 ¥14,000以上 ¥1,300
送料 ¥14,500以上 ¥1,340
送料 ¥15,000以上 ¥1,380
送料 ¥15,500以上 ¥1,420
送料 ¥16,000以上 ¥1,460
送料 ¥16,500以上 ¥1,500
送料 ¥17,000以上 ¥1,540
送料 ¥17,500以上 ¥1,580
送料 ¥18,000以上 ¥1,620
送料 ¥18,500以上 ¥1,660
送料 ¥19,000以上 ¥1,700
送料 ¥19,500以上 ¥1,740
送料 ¥20,000以上 ¥1,780
送料 ¥20,500以上 ¥1,820
送料 ¥21,000以上 ¥1,860
送料 ¥21,500以上 ¥1,900
送料 ¥22,000以上 ¥1,940
送料 ¥22,500以上 ¥1,980
送料 ¥23,000以上 ¥2,020
送料 ¥23,500以上 ¥2,060
送料 ¥24,000以上 ¥2,100
送料 ¥24,500以上 ¥2,140
送料 ¥25,000以上 ¥2,180
送料 ¥25,500以上 ¥2,220
送料 ¥26,000以上 ¥2,260
送料 ¥26,500以上 ¥2,300
送料 ¥27,000以上 ¥2,340
送料 ¥27,500以上 ¥2,380
送料 ¥28,000以上 ¥2,420
送料 ¥28,500以上 ¥2,460
送料 ¥29,000以上 ¥2,500
送料 ¥29,500以上 ¥2,540
送料 ¥30,000以上 ¥2,580
送料 ¥30,500以上 ¥2,620
送料 ¥31,000以上 ¥2,660
送料 ¥31,500以上 ¥2,700
送料 ¥32,000以上 ¥2,740
送料 ¥32,500以上 ¥2,780
送料 ¥33,000以上 ¥2,820
送料 ¥33,500以上 ¥2,860
送料 ¥34,000以上 ¥2,900
送料 ¥34,500以上 ¥2,940
送料 ¥35,000以上 ¥2,980
送料 ¥35,500以上 ¥3,020
送料 ¥36,000以上 ¥3,060
送料 ¥36,500以上 ¥3,100
送料 ¥37,000以上 ¥3,140
送料 ¥37,500以上 ¥3,180
送料 ¥38,000以上 ¥3,220
送料 ¥38,500以上 ¥3,260
送料 ¥39,000以上 ¥3,300
送料 ¥39,500以上 ¥3,340
送料 ¥40,000以上 ¥3,380
送料 ¥40,500以上 ¥3,420
送料 ¥41,000以上 ¥3,460
送料 ¥41,500以上 ¥3,500
送料 ¥42,000以上 ¥3,540
送料 ¥42,500以上 ¥3,580
送料 ¥43,000以上 ¥3,620
送料 ¥43,500以上 ¥3,660
送料 ¥44,000以上 ¥3,700
送料 ¥44,500以上 ¥3,740
送料 ¥45,000以上 ¥3,780
送料 ¥45,500以上 ¥3,820
送料 ¥46,000以上 ¥3,860
送料 ¥46,500以上 ¥3,900
送料 ¥47,000以上 ¥3,940
送料 ¥47,500以上 ¥3,980
送料 ¥48,000以上 ¥4,020
送料 ¥48,500以上 ¥4,060
送料 ¥49,000以上 ¥4,100
送料 ¥49,500以上 ¥4,140
送料 ¥50,000以上 ¥4,180
送料 ¥50,500以上 ¥4,220
送料 ¥51,000以上 ¥4,260
送料 ¥51,500以上 ¥4,300
送料 ¥52,000以上 ¥4,340
送料 ¥52,500以上 ¥4,380
送料 ¥53,000以上 ¥4,420
送料 ¥53,500以上 ¥4,460
送料 ¥54,000以上 ¥4,500
送料 ¥54,500以上 ¥4,540
送料 ¥55,000以上 ¥4,580
送料 ¥55,500以上 ¥4,620
送料 ¥56,000以上 ¥4,660
送料 ¥56,500以上 ¥4,700
送料 ¥57,000以上 ¥4,740
送料 ¥57,500以上 ¥4,780
送料 ¥58,000以上 ¥4,820
送料 ¥58,500以上 ¥4,860
送料 ¥59,000以上 ¥4,900
送料 ¥59,500以上 ¥4,940
送料 ¥60,000以上 ¥4,980
送料 ¥60,500以上 ¥5,020
送料 ¥61,000以上 ¥5,060
送料 ¥61,500以上 ¥5,100
送料 ¥62,000以上 ¥5,140
送料 ¥62,500以上 ¥5,180
送料 ¥63,000以上 ¥5,220
送料 ¥63,500以上 ¥5,260
送料 ¥64,000以上 ¥5,300
送料 ¥64,500以上 ¥5,340
送料 ¥65,000以上 ¥5,380
送料 ¥65,500以上 ¥5,420
送料 ¥66,000以上 ¥5,460
送料 ¥66,500以上 ¥5,500
送料 ¥67,000以上 ¥5,540
送料 ¥67,500以上 ¥5,580
送料 ¥68,000以上 ¥5,620
送料 ¥68,500以上 ¥5,660
送料 ¥69,000以上 ¥5,700
送料 ¥69,500以上 ¥5,740
送料 ¥70,000以上 ¥5,780
送料 ¥70,500以上 ¥5,820
送料 ¥71,000以上 ¥5,860
送料 ¥71,500以上 ¥5,900
送料 ¥72,000以上 ¥5,940
送料 ¥72,500以上 ¥5,980
送料 ¥73,000以上 ¥6,020
送料 ¥73,500以上 ¥6,060
送料 ¥74,000以上 ¥6,100
送料 ¥74,500以上 ¥6,140
送料 ¥75,000以上 ¥6,180
送料 ¥75,500以上 ¥6,220
送料 ¥76,000以上 ¥6,260
送料 ¥76,500以上 ¥6,300
送料 ¥77,000以上 ¥6,340
送料 ¥77,500以上 ¥6,380
送料 ¥78,000以上 ¥6,420
送料 ¥78,500以上 ¥6,460
送料 ¥79,000以上 ¥6,500
送料 ¥79,500以上 ¥6,540
送料 ¥80,000以上 ¥6,580
送料 ¥80,500以上 ¥6,620
送料 ¥81,000以上 ¥6,660
送料 ¥81,500以上 ¥6,700
送料 ¥82,000以上 ¥6,740
送料 ¥82,500以上 ¥6,780
送料 ¥83,000以上 ¥6,820
送料 ¥83,500以上 ¥6,860
送料 ¥84,000以上 ¥6,900
送料 ¥84,500以上 ¥6,940
送料 ¥85,000以上 ¥6,980
送料 ¥85,500以上 ¥7,020
送料 ¥86,000以上 ¥7,060
送料 ¥86,500以上 ¥7,100
送料 ¥87,000以上 ¥7,140
送料 ¥87,500以上 ¥7,180
送料 ¥88,000以上 ¥7,220
送料 ¥88,500以上 ¥7,260
送料 ¥89,000以上 ¥7,300
送料 ¥89,500以上 ¥7,340
送料 ¥90,000以上 ¥7,380
送料 ¥90,500以上 ¥7,420
送料 ¥91,000以上 ¥7,460
送料 ¥91,500以上 ¥7,500
送料 ¥92,000以上 ¥7,540
送料 ¥92,500以上 ¥7,580
送料 ¥93,000以上 ¥7,620
送料 ¥93,500以上 ¥7,660
送料 ¥94,000以上 ¥7,700
送料 ¥94,500以上 ¥7,740
送料 ¥95,000以上 ¥7,780
送料 ¥95,500以上 ¥7,820
送料 ¥96,000以上 ¥7,860
送料 ¥96,500以上 ¥7,900
送料 ¥97,000以上 ¥7,940
送料 ¥97,500以上 ¥7,980
送料 ¥98,000以上 ¥8,020
送料 ¥98,500以上 ¥8,060
送料 ¥99,000以上 ¥8,100
送料 ¥99,500以上 ¥8,140
送料 ¥100,000以上 ¥8,180
送料 ¥100,500以上 ¥8,220
送料 ¥101,000以上 ¥8,260
送料 ¥101,500以上 ¥8,300
送料 ¥102,000以上 ¥8,340
送料 ¥102,500以上 ¥8,380
送料 ¥103,000以上 ¥8,420
送料 ¥103,500以上 ¥8,460
送料 ¥104,000以上 ¥8,500
送料 ¥104,500以上 ¥8,540
送料 ¥105,000以上 ¥8,580
送料 ¥105,500以上 ¥8,620
送料 ¥106,000以上 ¥8,660
送料 ¥106,500以上 ¥8,700
送料 ¥107,000以上 ¥8,740
送料 ¥107,500以上 ¥8,780
送料 ¥108,000以上 ¥8,820
送料 ¥108,500以上 ¥8,860
送料 ¥109,000以上 ¥8,900
送料 ¥109,500以上 ¥8,940
送料 ¥110,000以上 ¥8,980
送料 ¥110,500以上 ¥9,020
送料 ¥111,000以上 ¥9,060
送料 ¥111,500以上 ¥9,100
送料 ¥112,000以上 ¥9,140
送料 ¥112,500以上 ¥9,180
送料 ¥113,000以上 ¥9,220
送料 ¥113,500以上 ¥9,260
送料 ¥114,000以上 ¥9,300
送料 ¥114,500以上 ¥9,340
送料 ¥115,000以上 ¥9,380
送料 ¥115,500以上 ¥9,420
送料 ¥116,000以上 ¥9,460
送料 ¥116,500以上 ¥9,500
送料 ¥117,000以上 ¥9,540
送料 ¥117,500以上 ¥9,580
送料 ¥118,000以上 ¥9,620
送料 ¥118,500以上 ¥9,660
送料 ¥119,000以上 ¥9,700
送料 ¥119,500以上 ¥9,740
送料 ¥120,000以上 ¥9,780
送料 ¥120,500以上 ¥9,820
送料 ¥121,000以上 ¥9,860
送料 ¥121,500以上 ¥9,900
送料 ¥122,000以上 ¥9,940
送料 ¥122,500以上 ¥9,980
送料 ¥123,000以上 ¥10,020
送料 ¥123,500以上 ¥10,060
送料 ¥124,000以上 ¥10,100
送料 ¥124,500以上 ¥10,140
送料 ¥125,000以上 ¥10,180
送料 ¥125,500以上 ¥10,220
送料 ¥126,000以上 ¥10,260
送料 ¥126,500以上 ¥10,300
送料 ¥127,000以上 ¥10,340
送料 ¥127,500以上 ¥10,380
送料 ¥128,000以上 ¥10,420
送料 ¥128,500以上 ¥10,460
送料 ¥129,000以上 ¥10,500
送料 ¥129,500以上 ¥10,540
送料 ¥130,000以上 ¥10,580
送料 ¥130,500以上 ¥10,620
送料 ¥131,000以上 ¥10,660
送料 ¥131,500以上 ¥10,700
送料 ¥132,000以上 ¥10,740
送料 ¥132,500以上 ¥10,780
送料 ¥133,000以上 ¥10,820
送料 ¥133,500以上 ¥10,860
送料 ¥134,000以上 ¥10,900
送料 ¥134,500以上 ¥10,940
送料 ¥135,000以上 ¥10,980
送料 ¥135,500以上 ¥11,020
送料 ¥136,000以上 ¥11,060
送料 ¥136,500以上 ¥11,100
送料 ¥137,000以上 ¥11,140
送料 ¥137,500以上 ¥11,180
送料 ¥138,000以上 ¥11,220
送料 ¥138,500以上 ¥11,260
送料 ¥139,000以上 ¥11,300
送料 ¥139,500以上 ¥11,340
送料 ¥140,000以上 ¥11,380
送料 ¥140,500以上 ¥11,420
送料 ¥141,000以上 ¥11,460
送料 ¥141,500以上 ¥11,500
送料 ¥142,000以上 ¥11,540
送料 ¥142,500以上 ¥11,580
送料 ¥143,000以上 ¥11,620
送料 ¥143,500以上 ¥11,660
送料 ¥144,000以上 ¥11,700
送料 ¥144,500以上 ¥11,740
送料 ¥145,000以上 ¥11,780
送料 ¥145,500以上 ¥11,820
送料 ¥146,000以上 ¥11,860
送料 ¥146,500以上 ¥11,900
送料 ¥147,000以上 ¥11,940
送料 ¥147,500以上 ¥11,980
送料 ¥148,000以上 ¥12,020
送料 ¥148,500以上 ¥12,060
送料 ¥149,000以上 ¥12,100
送料 ¥149,500以上 ¥12,140
送料 ¥150,000以上 ¥12,180
送料 ¥150,500以上 ¥12,220
送料 ¥151,000以上 ¥12,260
送料 ¥151,500以上 ¥12,300
送料 ¥152,000以上 ¥12,340
送料 ¥152,500以上 ¥12,380
送料 ¥153,000以上 ¥12,420
送料 ¥153,500以上 ¥12,460
送料 ¥154,000以上 ¥12,500
送料 ¥154,500以上 ¥12,540
送料 ¥155,000以上 ¥12,580
送料 ¥155,500以上 ¥12,620
送料 ¥156,000以上 ¥12,660
送料 ¥156,500以上 ¥12,700
送料 ¥157,000以上 ¥12,740
送料 ¥157,500以上 ¥12,780
送料 ¥158,000以上 ¥12,820
送料 ¥158,500以上 ¥12,860
送料 ¥159,000以上 ¥12,900
送料 ¥159,500以上 ¥12,940
送料 ¥160,000以上 ¥12,980
送料 ¥160,500以上 ¥13,020
送料 ¥161,000以上 ¥13,060
送料 ¥161,500以上 ¥13,100
送料 ¥162,000以上 ¥13,140
送料 ¥162,500以上 ¥13,180
送料 ¥163,000以上 ¥13,220
送料 ¥163,500以上 ¥13,260
送料 ¥164,000以上 ¥13,300
送料 ¥164,500以上 ¥13,340
送料 ¥165,000以上 ¥13,380
送料 ¥165,500以上 ¥13,420
送料 ¥166,000以上 ¥13,460
送料 ¥166,500以上 ¥13,500
送料 ¥167,000以上 ¥13,540
送料 ¥167,500以上 ¥13,580
送料 ¥168,000以上 ¥13,620
送料 ¥168,500以上 ¥13,660
送料 ¥169,000以上 ¥13,700
送料 ¥169,500以上 ¥13,740
送料 ¥170,000以上 ¥13,780
送料 ¥170,500以上 ¥13,820
送料 ¥171,000以上 ¥13,860
送料 ¥171,500以上 ¥13,900
送料 ¥172,000以上 ¥13,940
送料 ¥172,500以上 ¥13,980
送料 ¥173,000以上 ¥14,020
送料 ¥173,500以上 ¥14,060
送料 ¥174,000以上 ¥14,100
送料 ¥174,500以上 ¥14,140
送料 ¥175,000以上 ¥14,180
送料 ¥175,500以上 ¥14,220
送料 ¥176,000以上 ¥14,260
送料 ¥176,500以上 ¥14,300
送料 ¥177,000以上 ¥14,340
送料 ¥177,500以上 ¥14,380
送料 ¥178,000以上 ¥14,420
送料 ¥178,500以上 ¥14,460
送料 ¥179,000以上 ¥14,500
送料 ¥179,500以上 ¥14,540
送料 ¥180,000以上 ¥14,580
送料 ¥180,500以上 ¥14,620
送料 ¥181,000以上 ¥14,660
送料 ¥181,500以上 ¥14,700
送料 ¥182,000以上 ¥14,740
送料 ¥182,500以上 ¥14,780
送料 ¥183,000以上 ¥14,820
送料 ¥183,500以上 ¥14,860
送料 ¥184,000以上 ¥14,900
送料 ¥184,500以上 ¥14,940
送料 ¥185,000以上 ¥14,980
送料 ¥185,500以上 ¥15,020
送料 ¥186,000以上 ¥15,060
送料 ¥186,500以上 ¥15,100
送料 ¥187,000以上 ¥15,140
送料 ¥187,500以上 ¥15,180
送料 ¥188,000以上 ¥15,220
送料 ¥188,500以上 ¥15,260
送料 ¥189,000以上 ¥15,300
送料 ¥189,500以上 ¥15,340
送料 ¥190,000以上 ¥15,380
送料 ¥190,500以上 ¥15,420
送料 ¥191,000以上 ¥15,460
送料 ¥191,500以上 ¥15,500
送料 ¥192,000以上 ¥15,540
送料 ¥192,500以上 ¥15,580
送料 ¥193,000以上 ¥15,620
送料 ¥193,500以上 ¥15,660
送料 ¥194,000以上 ¥15,700
送料 ¥194,500以上 ¥15,740
送料 ¥195,000以上 ¥15,780
送料 ¥195,500以上 ¥15,820
送料 ¥196,000以上 ¥15,860
送料 ¥196,500以上 ¥15,900
送料 ¥197,000以上 ¥15,940
送料 ¥197,500以上 ¥15,980
送料 ¥198,000以上 ¥16,020
送料 ¥198,500以上 ¥16,060
送料 ¥199,000以上 ¥16,100
送料 ¥199,500以上 ¥16,140
送料 ¥200,000以上 ¥16,180
送料 ¥200,500以上 ¥16,220
送料 ¥201,000以上 ¥16,260
送料 ¥201,500以上 ¥16,300
送料 ¥202,000以上 ¥16,340
送料 ¥202,500以上 ¥16,380
送料 ¥203,000以上 ¥16,420
送料 ¥203,500以上 ¥16,460
送料 ¥204,000以上 ¥16,500
送料 ¥204,500以上 ¥16,540
送料 ¥205,000以上 ¥16,580
送料 ¥205,500以上 ¥16,620
送料 ¥206,000以上 ¥16,660
送料 ¥206,500以上 ¥16,700
送料 ¥207,000以上 ¥16,740
送料 ¥207,500以上 ¥16,780
送料 ¥208,000以上 ¥16,820
送料 ¥208,500以上 ¥16,860
送料 ¥209,000以上 ¥16,900
送料 ¥209,500以上 ¥16,940
送料 ¥210,000以上 ¥16,980
送料 ¥210,500以上 ¥17,020
送料 ¥211,000以上 ¥17,060
送料 ¥211,500以上 ¥17,100
送料 ¥212,000以上 ¥17,140
送料 ¥212,500以上 ¥17,180
送料 ¥213,000以上 ¥17,220
送料 ¥213,500以上 ¥17,260
送料 ¥214,000以上 ¥17,300
送料 ¥214,500以上 ¥17,340
送料 ¥215,000以上 ¥17,380
送料 ¥215,500以上 ¥17,420
送料 ¥216,000以上 ¥17,460
送料 ¥216,500以上 ¥17,500
送料 ¥217,000以上 ¥17,540
送料 ¥217,500以上 ¥17,580
送料 ¥218,000以上 ¥17,620
送料 ¥218,500以上 ¥17,660
送料 ¥219,000以上 ¥17,700
送料 ¥219,500以上 ¥17,740
送料 ¥220,000以上 ¥17,780
送料 ¥220,500以上 ¥17,820
送料 ¥221,000以上 ¥17,860
送料 ¥221,500以上 ¥17,900
送料 ¥222,000以上 ¥17,940
送料 ¥222,500以上 ¥17,980
送料 ¥223,000以上 ¥18,020
送料 ¥223,500以上 ¥18,060
送料 ¥224,000以上 ¥18,100
送料 ¥224,500以上 ¥18,140
送料 ¥225,000以上 ¥18,180
送料 ¥225,500以上 ¥18,220
送料 ¥226,000以上 ¥18,260
送料 ¥226,500以上 ¥18,300
送料 ¥227,000以上 ¥18,340
送料 ¥227,500以上 ¥18,380
送料 ¥228,000以上 ¥18,420
送料 ¥228,500以上 ¥18,460
送料 ¥229,000以上 ¥18,500
送料 ¥229,500以上 ¥18,540
送料 ¥230,000以上 ¥18,580
送料 ¥230,500以上 ¥18,620
送料 ¥231,000以上 ¥18,660
送料 ¥231,500以上 ¥18,700
送料 ¥232,000以上 ¥18,740
送料 ¥232,500以上 ¥18,780
送料 ¥233,000以上 ¥18,820
送料 ¥233,500以上 ¥18,860
送料 ¥234,000以上 ¥18,900
送料 ¥234,500以上 ¥18,940
送料 ¥235,000以上 ¥18,980
送料 ¥235,500以上 ¥19,020
送料 ¥236,000以上 ¥19,060
送料 ¥236,500以上 ¥19,100
送料 ¥237,000以上 ¥19,140
送料 ¥237,500以上 ¥19,180
送料 ¥238,000以上 ¥19,220
送料 ¥238,500以上 ¥19,260
送料 ¥239,000以上 ¥19,300
送料 ¥239,500以上 ¥19,340
送料 ¥240,000以上 ¥19,380
送料 ¥240,500以上 ¥19,420
送料 ¥241,000以上 ¥19,460
送料 ¥241,500以上 ¥19,500
送料 ¥242,000以上 ¥19,540
送料 ¥242,500以上 ¥19,580
送料 ¥243,000以上 ¥19,620
送料 ¥243,500以上 ¥19,660
送料 ¥244,000以上 ¥19,700
送料 ¥244,500以上 ¥19,740
送料 ¥245,000以上 ¥19,780
送料 ¥245,500以上 ¥19,820
送料 ¥246,000以上 ¥19,860
送料 ¥246,500以上 ¥19,900
送料 ¥247,000以上 ¥19,940
送料 ¥247,500以上 ¥19,980
送料 ¥248,000以上 ¥20,020
送料 ¥248,500以上 ¥20,060
送料 ¥249,000以上 ¥20,100
送料 ¥249,500以上 ¥20,140
送料 ¥250,000以上 ¥20,180
送料 ¥250,500以上 ¥20,220
送料 ¥251,000以上 ¥20,260
送料 ¥251,500以上 ¥20,300
送料 ¥252,000以上 ¥20,340
送料 ¥252,500以上 ¥20,380
送料 ¥253,000以上 ¥20,420
送料 ¥253,500以上 ¥20,460
送料 ¥254,000以上 ¥20,500
送料 ¥254,500以上 ¥20,540
送料 ¥255,000以上 ¥20,580
送料 ¥255,500以上 ¥20,620
送料 ¥256,000以上 ¥20,660
送料 ¥256,500以上 ¥20,700
送料 ¥257,000以上 ¥20,740
送料 ¥257,500以上 ¥20,780
送料 ¥258,000以上 ¥20,820
送料 ¥258,500以上 ¥20,860
送料 ¥259,000以上 ¥20,900
送料 ¥259,500以上 ¥20,940
送料 ¥260,000以上 ¥20,980
送料 ¥260,500以上 ¥21,020
送料 ¥261,000以上 ¥21,060
送料 ¥261,500以上 ¥21,100
送料 ¥262,000以上 ¥21,140
送料 ¥262,500以上 ¥21,180
送料 ¥263,000以上 ¥21,220
送料 ¥263,500以上 ¥21,260
送料 ¥264,000以上 ¥21,300
送料 ¥264,500以上 ¥21,340
送料 ¥265,000以上 ¥21,380
送料 ¥265,500以上 ¥21,420
送料 ¥266,000以上 ¥21,460
送料 ¥266,500以上 ¥21,500
送料 ¥267,000以上 ¥21,540
送料 ¥267,500以上 ¥21,580
送料 ¥268,000以上 ¥21,620
送料 ¥268,500以上 ¥21,660
送料 ¥269,000以上 ¥21,700
送料 ¥269,500以上 ¥21,740
送料 ¥270,000以上 ¥21,780
送料 ¥270,500以上 ¥21,820
送料 ¥271,000以上 ¥21,860
送料 ¥271,500以上 ¥21,900
送料 ¥272,000以上 ¥21,940
送料 ¥272,500以上 ¥21,980
送料 ¥273,000以上 ¥22,020
送料 ¥273,500以上 ¥22,060
送料 ¥274,000以上 ¥22,100
送料 ¥274,500以上 ¥22,140
送料 ¥2

マルゼンクレジット

- インテル8085CPU採用
- 5V単一電源で作動
- アセンブリ言語入力ボード付(タッチキー方式)ワード単位(8ビット)でダイレクト入力
- ビデオ・インターフェース(オプション)をつければ、すぐ画像が出せます。

頭金 ￥34,800
初回 ￥8,150
2～6回目 ￥8,000

INPEC-85AP ￥79,800



月々8000円でマイコンを…

TK-80	頭金 ￥13,000	初回 ￥8,300	2～10回目 ￥8,000
TK-80E	” ￥22,000	” ￥8,150	2～6回目 ￥8,000
TK-80BS	” ￥43,000	” ￥7,200	2～12回目 ￥8,000
LKIT8	” ￥12,000	” ￥8,300	2～10回目 ￥8,000
LKIT16	” ￥13,000	” ￥7,200	2～12回目 ￥8,000
H68/TR	” ￥14,500	” ￥7,200	2～12回目 ￥8,000
MK-80A	” ￥23,000	” ￥8,150	2～6回目 ￥8,000

支払回数・頭金・ボーナス利用等詳しい事は下記へお問合わせ下さい。

マイコン：上記の他ナショナルバナキットKX-33他

三菱、ナショナルセミコンダクター等各社製品。

電源：TOKTRM003(+5V10A、+12V1A、-5V1A)、RM05-06S(+5V10A) 日豪NPR-3M10(+5V3A、+12V0.5A、-5V0.5A) NPR3M50(+5V5A、+12V0.5A、-5V0.5A)他。

測定器：LEADERシクロスコープLB0-508(130%、20MHz、10mV/cm 2現象)他。

トリオ、菊水等各社製品。

ハンダゴテ：Ungar #127(3線式24W)他。

その他：TTL・DTL ICのテストに最適なLED使用スタンレーロジックチェッカー
ソルダーヘルパー・精密ラジオペンチ・ニッパー等エンジニアの工具。

本：マイコン関係月刊紙(新刊・バックナンバー)他 各種。



システム・フロア 本社ビル2F

電子のキャンパス

丸善無線電機株式会社

東京都千代田区神田佐久間町1丁目8番地

☎ 255-4911(代表)

「ミスプリントではありません」

新発売 タイプライター

¥99,800

TP-038A

現品先渡し=後払い!



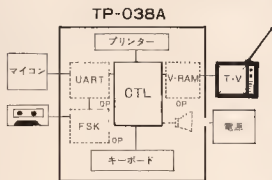
特許出願中・実用新案出願中

お支払い方法	アンツ タイプライター	
	TP-038A	TP-035A
分割払金	6,000円	10,500円
支払回数/支払期間	20回/20ヶ月	20回/20ヶ月
分割払価格/(合計)	108,000円	192,000円
現金価格(一括払)	99,800円	178,000円

Ants[®] TIEPRINTER

業務仕様製品をあなたのマイコンのI/Oとして

アマチュア価格で特別提供 ASR-33モデルベース



TP-038A

仕様

- ・キーボード: ASCIIフルキーボード
- ・プリンター: 放電式5×7ドットプリンター
- ・印字: 64キヤラクター
1行32、21、桁印字可能
- ・送受信: 全二重
- ・入出力: パラレル
- ・ローカル機能有り/マイコン無接続でも1字づつタイプライターとして印字出来ます
- ・寸法: 470W×210D×90Hmm
- ・形状: 卓上型

■TP-035A 仕様 ¥178,000

- ・ASR-33モデルベース
- ・入出力: シリアル
- ・FSK: カセットインターフェース内蔵
- ・リピーター機能内蔵
- ・印字音、出力機能内蔵

■TP-036A(J) 仕様 (受託生産品)(別途価格)

- ・マイコンシステムメンテナンスツール
- ・形状: アタリシユゲース型
- ・ASR-33 モデルベース
- ・カセットメカ内蔵
- ・AC100V用電源内蔵

TP-038A型タイプライターは、TP-035A・TP-036A(J)型をモデルとし特にお求めやすい価格に改良を加え破格の超低価格でお求め頂けるはこびとなりました。あなたのマイコンライフにより強力な生命をTP-038Aは、お約束出来ます。

電子技術社員募集30才まで経験者優遇
営業社員募集20才～30才まで経験者優遇
お気軽に電話ください。店員の秘密は厳守します。

お問い合わせは下記へ
販売店募集中

Ants[®]
ELECTRONICS INSTRUMENTS

発売元

アンツ 電子機器販売(株)

〒461 名古屋市中区東横2-3-7(東カンビル)
TEL (052) 932-1720代

製造元

株式会社 **アンツ**

〒501-04 岐阜県本巣郡真正町軒海313の1
TEL (0583) 24-3167代

技術の粋 多様な応用

MM80-4K/1K ¥27,500 千300

MM-16K/4K4K ¥69,500 千300

AD-08KIT ¥15,500 マニュアル付

MS9026K ¥4,800 マニュアル付

Dlog-08KIT ¥24,500 千300

AD-203KIT ¥13,500 千300

MS-202KIT ¥7,500 千300

タイムプロセッサ TP-2

マイコン内蔵のプログラマタライザ

TP-2C (ケース入り、完成品) ¥48,500

TP-2B ICPUポート拡張キット ¥33,500

TP-2K ICPUポートTP-2B付 ¥29,500

特徴

● 8080/8085同期方式

● 1分間・1ミリの時間測定

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

● 100V、50Hzの出力

NEC μCOM Basic Station

TK-80BS ¥128,000

専用電源 IC-0005 ¥23,500 千1,000

電源 IC0004 ¥9,800 千500

LKIT-16 ¥98,000

LN1630 ¥6,000 HB1111 ¥900 μB515 ¥9,000

LA05KA(テレビンタフェース) ¥93,000

LA05KB(カダラフティックオプション) ¥29,000

LA05KD(カセットMTテレタイプインタフェース) ¥17,500

H68TR ¥99,000

HM47214-4 ¥3,500

H6T V01(TVインタフェースモジュール) ¥69,500

EX80 ¥85,000

E-PACS II

ユニバーサル基板

●特徴

1) ランディング実装

2) 100P-42PまでのICを実装可能

3) 動作・研究用としても優秀

4) 変換できる

4×4×H/D□□ 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175×220 120mm

LB1405キット ¥1,980

(レベルメータキット)説明書付

VCC4.4V~12V

VIN-0.3~14V

LB1405×2+基板表示器+CR

●特徴

1) 100P-42PまでのICを実装可能

2) 動作・研究用としても優秀

3) 変換できる

4) 変換器、ガラス封入

IC最大消費電力(100V時)

430 IC42NE ¥5,000 ¥4,000 125×220 7mm

421 IC42NE ¥7,200 ¥4,500 125×100 110mm

421 IC42NE ¥9,500 ¥3,500 175

HAMERS

2-7-21, MATSUKAGE-CHO, NAKA-KU, YOKOHAMA
231 JAPAN TELPHONE:045(662)0688

ハマーズ

☆全国クレジット取扱中

○中越から北海道まで全国分館OKノ
○当社の取扱全製品(3万円以下)
○申し込みは電話でOKノ

○1~24回払い(ボーナス借付可)
○頭金ナシでもOKノ
○おまけに格安

HEATH KIT 完成品
H8 cpu+H9 video
¥487,000



(H8, H9)
(例) 現金 0円
1回目 26,696円
2~24回 25,000円×23

コモドール
PET 2001
¥298,000千2,000



(例) 現金 0円
1回目 23,184円
2~24回 15,000円×23

アドテック
ADB002 実装 TVTY-1
完成品 ¥168,000千番払



(例) 現金 0円
1回目 12,067円
2~24回 8,500円×23

Sunpec
8000TK+TK80E
¥185,600 +8000power



(8000TK)
(例) 現金 0円
1回目 10,812円
2~24回 9,500円×23

NEC
TK80E
¥67,000千サービス



(例) 現金 0円
1回目 7,215円
2~12回 6,200円×11

日立
H68/TR
¥99,500千サービス



(例) 現金 0円
1回目 9,697円
2~12回 9,300円×11

パナファコム
LKIT-16
¥98,000千サービス



(例) 現金 0円
1回目 9,108円
2~12回 9,200円×11

パナファコム
LKIT16+CRT
¥166,300 +マザーボード
+3M50



(拡張ボード)
(例) 現金 0円
1回目 9,966円
2~24回 8,500円×23

NEC
TK80E+TK80BS+3M50
¥212,500



(TK80BS)
(例) 現金 0円
1回目 9,548円
2~24回 11,000円×23

日立
H68/TR+H68/TV
+5V10A
¥197,000(SWL0510)



(H68/TV)
(例) 現金 0円
1回目 13,397円
2~24回 10,000円×23

インターフェース
NEC TK80BS ¥128,000千サービス
日立 H68/TV ¥69,500

パナファコム
メモリーボード ¥42,000
CRT 39,000
プリンターI/O ¥24,500
カセット ¥17,500
マザーボード ¥11,800
カテゴリー ¥29,000
RFモジュール ¥2,000
放電プリンター ¥16,000

周辺装置
ディックデジタルカセット+MT2
¥95,000



TSP7706A 放電プリンター
¥37,000千サービス

Power Supply

日立 HTP-503(5V, 3A) H68/TR専用PS ¥11,500 千500
日立 NPR 3M10(+5V3A, 5V0 3A, +12V0 5A) ¥12,500 千サービス
日立 NPR 3M20(+5V3A, 12V0 5A, +12V0 5A) ¥12,500 千サービス
日立 NPR 3M50(+5V3A, 5V0 5A, +12V0 5A) ¥17,500 千サービス
日立 NPR 3M60(+5V3A, 12V0 5A, +12V0 5A) ¥17,500 千サービス
日立 SWL-0510 スイッチングPS(5V, 10A) ¥26,000 千サービス

★売場拡張し電子パーツ(CR, Tr, IC又拡張用RAM, ROM等)の販売を行なっております。又売場内にplay room (マイコン等)を増設、自由に御使用下さい。

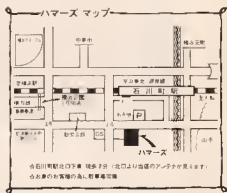
★通信販売の場合は現金書留又はお振込口座三菱銀行上野支店当座No9005468へお振込み下さい。

★クレジット申し込みの注意
申し込み者が20才未満又学生の方の場合は保護者の方を申し込み者にして下さい。

工人舎

横浜市中区松影町2-7-21
〒231 ☎045-662-0688(代)
営業時間AM10:00~PM7:30

定休日 毎週水曜日(祭日の場合は翌日)



HAMERS クレジット 申込書		商 品 名		お 支 払 回 数		6・10・12・16・18・20・24回	
名 前		姓	生年月日	年 月 日	才 電 話	— —	
住 所					居住年数	年	配偶者 有・無
お 勤 め 先					お 勤 め 年 数	年 月 収	万円
そ の 住 所					ご 住 居	自己所有・家族所有・借家・家・社宅・アパート	

I/O 6502-01 ONE BOARD COMPUTER

特徴

- 1Kキレタイプモーター (70M)
- テレタイプ用自動同期 (100-300ボートリアルレート)
- 8BIT バレタイ
- 8BITパラレルI/Oポート
- HIGH SPEEDプログラム
- 補正なし10進演算
- ミニコンに近いアドレス
- 完全スタック動作
- 内装4-AM (メモリ、ROM、I/O、ユーザー)
- DC8Vの直流電源のみで作
- 4K RAM実装時 TINY BASIC等 8BIT SUPOR各種用意

完成品¥39,000 キット¥34,800

仕様

- CPU 6502 (アッパル系同じ)
- MONITOR TM6530-004
- タロシタ (8-12MHz)
- RAM 2K14TYPE X2, 74LS245 X2, 74LS04 X2, 74LS10, 74LS00付
- BUFFER IC 74LS245 @ 515, 74LS365 @ 165
- PORT ターミナル (近日発売)
- ITY 形式 48 X 15 出力カレントはカレントループ (20mA)

● 56PINBUS 各種CPU基板 (全130 X 150mm)

- 6800 使用IC 74365 X3, 74LS245, 74LS10, 7405等の他
6802 使用IC 74365 X3, 40P, 16P,
ICソケット付、金メッキコネクタ、
エポキシ樹脂のみ ¥9,800
ICソケット付動作試験済 ¥12,800
6502 使用IC 74365 X3, 74LS245, 6530-004, 2114 X8
その他 基板のみ ¥12,000
ICソケット付、動作試験済 ¥14,800

● 紙テープリーダ駆動用ハイブリッド回路

パルスモータ駆動駆動速度可変、反転、停止、機構内蔵
¥3,800

● 日本航空電子製カードリーダMR-148型

8インチ、動作回数1,000,000回、テープホルダ付
¥110,000

● 日立フロッピーディスクFDD101A型

モータ駆動回路キット付 デスキット ¥238,000
¥3,000

BASFミニフロッピー6106

● 日本サーボ、サーボモータ、パルスモータ各種

● 日本電子ミラセラムハニカムヒーター

真空、乾燥機等のマイコン制御に最も安全な発熱体
TC190°C100V、200V用 最大出力350W 電圧付 ¥2,000

● I/O万能基板

56P 寸法130 X 150 半田メッキ、ピン金メッキ
ベーク ¥3,200
コート ¥4,000

お願い製品により納期のかかるものもありますので御注文の際には御一報下さい。

I/Oラボラトリー

〒101 東京都千代田区神田佐久間町1-14
☎ 03-251-5102 第2東ビル
〒185 東京都国分寺市本町4丁目21の8
☎ 0423-21-6650

新装開店

マイコン部門強化!!
ニュー秋葉原センター店

『フレッシュショップ』
 ツクモ5号店 秋葉原にオープン!

パートI TB-80 μ COMキット

マイコンマニアにおくる超コストパフォーマンスマシン

好評! 定価 ¥68,000 (ソフト TK-80 コンパチ)

- 1000AF使用電源は1-3Vですが、DC-DCコンバータを使用しておよそ5V、7V、8V、10V、12V、15Vの電圧が得られます。
- キーボード5Wは、リモートボードで、40mmの15ピン平行ケーブルで接続されています。



台数限定 ラストチャンス! **特価 ¥53,000** 千サービス

好評発売中! 2度とないチャンス!

MK-80A μ COMキット

(TK-80相当機)

台数限定特価 ¥63,000

千サービス

MK-80E ¥54,000

(TK-80相当機) 千1,000

TK-80BS・BS用ケースと合わせてお買求めください。

TK-80/MK-80A/TB-80用

高性能電源 TF-80

5V・1.5A、+12V・0.2A連続使用可

特価 ¥12,000 千500

- ロゴス8Kメモリボード S-100バス ¥16,000
- 4Kメモリボード ¥12,000
- 大版1CMキャリッジネポート ¥15,000
- 完全キット ¥37,000
- 完成品 ¥40,000
- 松久 MK型キーボードエンコーダなし ¥23,000
- エンコーダあり ¥35,000

各千サービス

8085時代のとびらを開く!

INPEC-85シリーズ

インテル8085 基本システムINPEC-85A

¥55,000

千サービス



専用電源 5V:2A PS-1 ¥8,000 千サービス

★各社 μ COMキット特価販売中!

- お問合せ下さい
- イムサ μ 8025基本システム ¥310,000
 - アップルIIスタンダード ¥245,100
 - TK-80BS ¥129,000
 - TK-80 ¥38,500
 - TK-80E ¥67,000
 - H83 TR ¥59,800
 - LK11-18 ¥35,000
 - LK11-18 ¥35,000
 - EK-80 ¥35,000

店頭販売のみ

- 2132AL-4(NEC) 特価 ¥380
- 2101AL-4() 特価 ¥700
- 8080A () 特価 ¥4,200
- M58751P(2102) 特価 ¥370
- M58721P(2101) 特価 ¥590
- HM472114 特価 ¥2,700

★マイコン関係のお問合は ニュー秋葉原センター店 ☎03(251)0986-8
 ラジオセンター店 ☎03(251)2637 名古屋店 ☎052(263)1635-6へどうぞ



九十九電機株式会社

- 万代店 幸10 東京都千代田区外神田1-3-9 ☎03(251)2441-3
- ニュー秋葉原センター店 幸10 東京都千代田区外神田1-6-10 ☎03(251)0986-8
- ラジオセンター店 幸10 東京都千代田区外神田1-4-2 ☎03(251)2637
- 定休日 毎週木曜日、第3水曜日 通話は万代店1/6係へ



名古屋店は上り前津駅が便利!

パートII 好評発売中!

オリジナルケース 大好評です。注文殺到! システムアップを考えて余格の電源設計です。

TF-80BSC デラックス

高性能10A電源 産業用マイクロファンを搭載! なんと **¥49,800** スペックアップしました。電源の価格とくらべてみて下さい。 千1,000



取付加工
 基盤はグレーの
 メタリック



モニターTVをのせます。

バスラインの穴

発売開始!

松久 MK型のアダプターも有ります(別売)

- TF-80BSC(3.5A)..... 特価 **¥39,800**
- TF-80BSCケース(ファン、SW付)..... 特価 **¥26,800**
- TF-80BSCケース(松久)MK型タイプ..... **¥27,800**
- 10Aユニット電源..... **¥24,800**

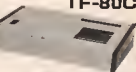
好評発売中!

TF-80C

側面は天然木材使用の
 グッドデザインケース
 TK-80/TK-80E

MK-80A用

オリジナル高級金属ケース



特価 **¥7,800 千700**

組合わせ特別価格の一例

- TK-80E システムでどうぞ
- TK-80BS **¥232,000 千落払**
- TF-80BSC
- ★TK-80BSお買上げの方に
 S1プロダクト用ケーブルを
 プレゼントいたします
- MK-80A
- TK-80BS **¥222,000 千落払**
- TF-80BSC

サンペックBASICシステムのインタープリタが— 実に強力になりました!!—



第二版サンペックの

ベーシックインタープリタの新機能

	コマンド	省略形	機能
キャラクターモード	●キャラクターモード $0 \leq a \leq 15, 0 \leq b \leq 31$		
	PEEK (a,b)	PE. (a,b)	(a,b) の値を関数としてもちがえる
	POKE (a,b)	PO. (a,b)	
	POKE (a,b) X		Xの値を代入する
	POKE (a,b) X		Xの値を5桁うちだす
	POKE (a,b) "..."		引用 "...内をそのままうちだす
	POKE (a,b) \$hhhh		hhhhに対応するASCIIをうちだす(カナ文字可能)
グラフィックモード	●グラフィック $0 \leq a \leq 46, 0 \leq b \leq 63$		
	CURS (a,b)	CU. (a,b)	カーソル移動
	ERASE (a)	E. (a)	$0 \leq a \leq 16$ a=16なら a行を消去 a=16なら E. と同じ
	ERASE	E.	全画面消去 (キャラクターモード)
サウンド	●グラフィックモードのグラフィックモードのPEEKに相当		
	PICK (a,b)	PI. (a,b)	
	PLOT (a,b)	PL. (a,b)	
	PLOT (a,b)		(a,b) にプロット (白トット)
	PLOTB (a,b)		(a,b) にプロット (黒トット)
	PLOTI (a,b)		(a,b) を反転する
	EG.		グラフィックモードの画面消去
機械語	BELL ()	B. ()	引数なしなら中央音を。そうでなければ
	BELL	B.	引数に応じた高さの音を出す (100m sec)
	SON		キー入力に対し確認用の音を出す。出さないかを指定する。
	SOFF		スイッチコマンド
	FNCn ()		nは0~9。特定番地への数をリンク。関数として値をもちがえる
	CALL { }	CA. { }	機械語ルーチンへのリンク
	PAUSE X	PA. X	プログラムの実行を中止する。多機能コマンド
機械語	KEY	K.	キー入力チェック
	RMZ ()	RM. ()	乱数の初期化

(今後はカナ文字のプリント文庫の発行も、機能向上のため進めさせていただきます)

このVer.2の作成にあたって、東山マイクロクラブのご協力をいただきました。(本山版Ver.6.75)

好評発売中!!

SUNPEC BASIC SYSTEM
MODEL SUNPEC 8000-TK

新インタープリタ付属

¥99,800 円
(¥1000)



写真はフルシステム

システムラックキット

新インタープリタ付属

¥226,000 円
(¥800)

- SUNPEC 8000シリーズ好評発売中
- 8000-TK BASICシステムキット ¥99,800 (¥1000)
 - 8000-04TK BASICラックキット ¥26,000 (¥700)
 - 8000-01 CRTディスプレイボード ¥37,000 (¥500)
 - 8000-02 4Kバイトメモリーボード ¥39,800 (¥500)
 - 8000-03 Fdiskカセット / ¥6,800 (¥350)
 - 8000-02B メンリーポート基板 ¥9,000 (¥350)
 - 8000-FAN 強制空冷キット ¥6,000 (¥500)
 - 2376BK 松久MF用エノコダーP板 ¥800 (¥200)

お問い合わせ、お買い求めは
旭川 山本電子、山形 ミニタツ、七巻街、東京 亜電電子、横浜
エー・エー、浜松 エー・エー電子工業、名古屋 カトー無線エー・エー、京都
エー・エー、大阪 日立電子産業、富山 豊橋、上新電機、神戸
星家パナ、姫路 星家パナ、岡山 エー・エー電子、広島
産業、北九州 北九州無線、大分 トキハマルト(オーディオ)、熊本
福岡電気商会、長崎 長崎電気商会、福岡 カホ、平塚、鹿児島
昭栄堂、宮城 昭栄堂

SUNPEC 総発売元

サン・エレクトロニクス・デザインセンター

〒483 愛知県江南市安良715 ☎05875-4-7111

SUNPEC 製造元/通信販売部

株式会社 マイクロシーエー技研

〒470-11 愛知県豊明市新田町子持松11-8 ☎0562-93-3118

大宮マイコンクラブ

—順調にスタート—

宮永好道先生(会長)指導による

企画の面白さ
指導の徹底

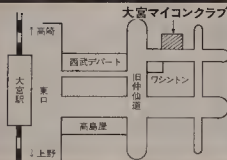
—会員募集中—

大宮マイコンクラブ 同マイコンショップ

埼玉県大宮市宮町2-7 (川崎屋ビル4F)
TEL 0486(44) 0904

後援 立之システムリサーチ株式会社

本社 埼玉県大宮市吉敷町4-33
TEL 0486(44)6125 (東芝ビル1F)



気軽に買える信頼のデバイス専門店

主な取扱い品種

□ マイクロコンピュータKIT
・MEK6800D-II・A(モトローラ)
・TK-80(NEC)

・L-KIT-8(富士通)
・L-KIT-16(パナファコム)
□ ナショナル家電プリンター
(21所, 32所, 40所)

□ インターフェース基板
1コントローラドライバ回路

□ マイコン用チップ

・μPD8080A 8Bit 8080A 8085A 8088A
・μPD8255C E プログラマブル周辺インターフェース
・μPD5101E フルデコード256×4Bit スタックRAM
・μPD454D 256W×8Bit EEPROM
・μPD472D 5120Bit キャラクタージェネレーター

・μPD2101 フルデコード256×4Bit スタックRAM
・μPD2102 フルデコード1024Bit スタックRAM

・μPB8212D 8Bit 1047ポート
・B8216D 4Bit 最大方向バス・ドライバ
・B8224D クロックジェネレーター
・μPB8228D システムコントローラー
・2513 キャラクタージェネレーター
(16, 25, 32)

□ 中, CMOS, 500シリーズ各種
4桁 BCD DECADE COUNTER
4桁 BCD DECADE COUNTER
・TC5011C (4DIGIT DECADE COUNTER)
……東芝
・TC5010P (ラッチ付, UP/DOWN COUNTER)
……東芝
・MSM5502 (4DIGIT DECADE COUNTER)
……沖

□ ラジオ周波数カウンタ
・M54821 (5DIGIT FREQUENCY COUNTER) ……三菱

□ 水晶
・1MHz (HC 6/u), 100KHz (HC 13/u)

□ レベルメータ用
・LB1405 (5個のLEDによって入力レベルを棒状に表示) ……三洋

□ 簡易形A/D変換器
・MS1901P (12点LEDドライバ)
……三菱

□ 各種 Operational Amplifiers
(例) ……741CP ⑧ ¥120 (10ヶ ¥1,000)

□ ボルテージレギュレーター
□ その他いろいろ特価販売中



各社IC半導体専門店

テクノカルサンヨー

〒556 大阪市浪速区日本橋4-1-17 豊岡ビル2F ☎(06)644-0785・(06)643-5209

※ 地方お送り即日発送。ご注文の際は、現金書留、又は「郵便為替」でお願いします。※ 代引もします。

華麗なるマイコン

APPLE II

——Color & High Resolutionの世界への招待——



Apple IIの特徴

- 15色のカラーグラフィック命令を持つ高速6K B A S I CがROMで装備されています。
- 10K、9桁の浮動小数点B A S I Cがテープで用意されていますから高度の演算にも応じられます。
- 強力なシステムモニタはAPPLE IIの機能をフルに発揮させますので、ソフトウェア作成が楽になります。
- ミニアセンブラ、ティスアセンブラ、16ビットマシン・シミュレーション、浮動小数点パッケージがROMで標準装備されたプロ級システムです。
- オーディオカセットインターフェイスは1500bpsと高速で、ローディングの時間は非常に短かくなっています。
- 軽量、小型の中にすべてが入っていますから、キャリングケースに入れてオフィスから自宅へ、友人宅へと持ち運びが簡単です。
- シリアルプリンタ、パラレルプリンタ、デジタルカセット、フロッピーディスクなどの周辺機器へ容易に接続可能です。
- 多点のゲーム用I/Oやスピーカーが装備され、容易にプログラム上で使用できますので応用がぐっと広がります。
- 280×192点4色で構成される高分解能グラフィックス・ルーチンは夢の世界を映像化してくれます。
- 購入元の棚イーエスティ・ラボラトリーは完全にAPPLE IIをサポートする技術力を持っています。

APPLE II 仕様

■ 寸法 387mm W × 457mm D × 133mm H

■ マイクロプロセッサ 6502 1MHzクロック

■ ビデオ・ディスプレイ

テキスト、カラーグラフィックスおよび高分解能カラーグラフィックスで、グラフィックスの場合下の4行がデキストとして使える。モードはソフトウェアで選択でき、またディスプレイに使うメモリを2つにわけ、ソフトウェアでどちらかをえらべます。

■ テキスト・モード

- 40×24行
- 5×7 アパーケース
- ノーマル、逆転、フラッシング可
- ディスプレイ コントロールはROM化されている
- カーソル機能はすべてOK
- 1000CPSの高速表示

■ カラー・グラフィックス

- 40H×48Vの分解能または40×40+4行のテキスト
- 15色
- BASICに独特な命令あり

■ 高分解能グラフィックス

- 280H×192Vまたは280H×160V+4行のテキスト
- 黒、白、紫、緑の4色表示
- 8Kバイトを表示

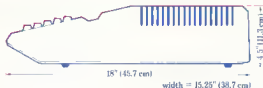
■ メモリ

4~48Kバイトまで同一ボードにRAMは増設可能。8KバイトのBASICおよびモニターROMは組込まれている。その他2つのROMソケットが付属。

■ I/O

- ASC11 キーボード
- オーディオ カセット 1500bps
- スピーカ
- ゲームI/O スティック入力×4、TTL入力×3
TTL出力×4
- ジョイスティック 2つ
- コネクタ 8つ バスラインバッファ、タイミング付、デジーチェーンのインタラプトおよびDMA優先度方式

寸法



■ BASIC

- 6KバイトのROM化した高速整数BASIC
- 変数名の長さは適当、例えばALPHA, EBTA\$
- 各種エラーが入力と同時に表示
- 複数の命令を同行内に書ける
- 整数は16ビット精度
- 文字列は255まで、D1Mはシングル可
- グラフィックス用命令 COLOR=exp, PLOT, HLIN (水平軸をかく), VLIN, SCR(x, y) (スクリーン上の色を読む)
- ジョイスティック用 PDL(0-3)
- BASICでテキストかグラフィックスかを指定
- メモリ境界アジャスト可能(プログラム駐在のまま)
- BREAK, CONTINUEあり
- DEBUG命令
- I/O選択可能
- DMA命令: PEEK, POKE, CALL
- カセット命令: SAVE, LOAD
- 自動行番号モードあり

■ モニタ

- スクリーン、カーソルコントロール
- スクロール、ウィンドウ可変
- シングル・ステップ、トレース
- 16ビット・プロセッサ・シミュレーション
- ディスアセンブラ/ミニアセンブラ
- I/O選択
- 浮動小数点パッケージ
- カセット・ルーチン
- ビデオ 正、反転セレクション
- 16進加減算 (相対番地計算)

周辺装置

● ハードコピー

TTY、各社放電プリンタ、パラレル入力プリンタ
SWTPC PR-40 ドット・マトリックス・インパクト
式印刷機R0-101P ドラム・インパクト(プロ用)
以上はすべてゲームI/Oスロットよりソフトウェアでドライブ可能です。

● マグネチックテープ

一般のオーディオカセット用インターフェイス
ディアン M7-2用インターフェイスカード

● フロッピーディスク

ミニフロッピー・ディスクドライブ及びソフトウェア76/151Kバイト
フルサイズ フロッピーはシグマート社または同等品を予定中

● 各種インターフェイス・ボード

● リアルI/O RS232C/20mAカレントループ ● パラレルI/O
入力8ビット、出力24ビット ● パラレルプリンタ用I/O ● シリアルボード

● インターバル・タイマ

6522使用 16ビットタイマ、インタラプト、16ビット入、出力ポート

APPLE IIには2種のカラーBASICが用意されています

電源ONで、すぐBASICを楽しみたい方は6K

●電源ONですぐにBASICを楽しみたい方はROMに入っている標準装備の6KカラーBASICをお使い下さい。RESET、コントロールCと2回のキー操作でOK。整数高速型で豊富な命令を持っていますから、15色のカラーグラフィックスやハイリゾリューショングラフィックスを含むプログラムを高度なものとして行うことができます。

6K BASICの命令

AUTO CALL CLR COLOR CON DEL DIM DSP END FOR TO STEP GOSUB GOTO GR HMMEM HLIN IF THEN INPUT IN LET LIST LOAD LOMEM MAN NEXT NEW NO DSP NOTRACE PEEK PLOT POKE POP PRINT PR REM RETURN RUN SAVE TAB TEXT TRACE VUN VTAB	
AUTO : 行番号自動発生	TRACE : 実行番号表示(デバッグ用)
CON : 実行モード	VTAB : カソールの垂直位置指定
DEL : 削除	GR : グラフィックス指定
DSP : 変数変化の表示(デバッグ用)	COLOR : 色の決定
IN : 入力ポート番号指定	HLIN : 横線の発生
LOAD : カセットテープよりのロー	PLOT : 点の発生
MAN : AUTO命令クリア	VLIN : 縦線の発生
PR : 出力ポート番号指定	SCRN : 色の値の読み出し
POP : スタックをのぞく	PDL : ゲームパッドの読み出し
SAVE : カセットテープへの書き込み	LEN : ストリング変数の長さ
TAB : カソールの水平位置指定	SGN : 符号発生
TEXT : 画面テキストモード指定	
OPERATOR : +, -, *, /, ^, <, >, !, # (NOT EQUAL), MOD (除算の余り), AND, OR, NOT	
FUNCTION : ABS, ASC, LEN, RND, SGN	

強力なモニタ

1. メモリ内容のチェック
2. メモリ内容の変更
3. メモリ内容のブロック移動
4. メモリ群の照合
5. オーディオカセット入出力ルーチン
6. 表示の正、反転
7. 命令の実行
8. 命令のトレース
9. シングルスステップモード
10. レジスタ群の表示と変更
11. サブルーチンのみの実行
12. 16進の加減算
13. 入出力ポートの設定
14. 効果的複合命令
15. フルカラー機能
ホーム、上、下、左、右移動可能、カソール以下1行消去
カソール以下全行消去
16. ウィンドスクローリング機能
スクローリング部の左端、横端、上端、縦端の設定
17. スピーク駆動
18. バドルセンス 4
19. スイッチセンス 3
20. アナウンシエタ出力 4
21. カラーグラフィックスセット
22. テキストモードセット
23. ミクスドモードセット
24. 第2ページ表示
25. 高分解能グラフィックスモードセット

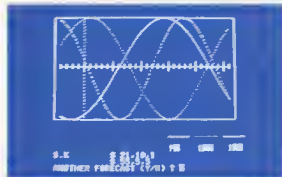
プロフェッショナルの方には

10K

●高精度の浮動小数点演算や各種の関数を必要となる方には10KフローティングBASICがあります。9ケタの精度を持っているこのBASICは更に命令が強化されていますし、関数も種類が多いので、科学技術計算や商業上の利用にもよいものです。

10K BASICの命令

6K BASIC 以外に DATA, DEF FN, ON GOTO, ON GOSUB, READ, RESTORE, STOP, ? CHAS, LEFT\$, MID\$, GET, RIGHT\$, STR\$, VAL, FUNCTION ATN, COS, EXP, FRE, INT, LOG, POS, SGN, SIN, SQR, TAN.



ディスペンサ

基地設定後Lをキーインすることによって、20ステップごとのディスペンサをなし、基地、機械語、アセンブリ言語をディスプレイする。

ミニアセンブラ

基地設定後アセンブリ言語で入力したプログラムを機械語に翻訳し、基地、機械語、アセンブリ言語を表示する。但しラベルはつけられません。

高分解能グラフィックスサブルーチン

- 1 NT : イニシャライズ
- CLEAR : 画面のクリア
- PLOT : 点の発生
- POS : 点の位置指定
- LINE : 線の発生
- SHAPE : あらかじめ定められた形の発生

16ビット マシン シミュレーション

Return to 6502 MODE, Branch, BCC, BCS, BPL, BMI, BEQ, BNE, BRX, NOP, IMMEDIATE, MOVE Registers, COMPARE, ADD, SUBTRACT, INCREMENT, DECREMENT.

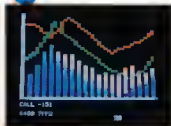
各種応用

各種ソフトウェア開発用、制御用、計画機器及び理化学測定機器データ処理用
売上計算処理、在庫管理、経理、工場管理、得意先管理等

豊富なアプリケーション・ソフト



ビジネスマンに



売上げ、利益の管理
経営の管理はディスク
を用いたファイルシス
テムにより、キーひと
つで表示されます。



小切手帳の集計
番号、宛先、項目等が
データベースできわめ
て容易になります。



エンジニア・研究者に



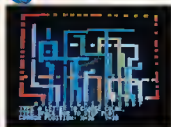
分布の色別表示
豊富な色を使っている
データの整理は、パター
ン認識の上できわめて
効果的です。



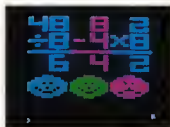
高分解能グラフィクス
高速データ取り込みと
演算、表示はCRTデ
ィスプレイによるカラ
ーならではの応用です。



お子様の教育に



ダズラ
15色のカラーチョーク
で自由自在に絵を描く
ことができます。お子
様はジョイスティックで
プロはフントウェアで
チャレンジしましょう。



カラーマス
楽しい計算、間違ると
こわい顔がにらみます。
さあもう一度やりナオ
シ。



ホビーに



スタートレック
あなたのエンタプライ
ズ号は超近代宇宙船。
ワープして空間をのり
こえクリンゴンを撃破
する壮大なアドベンチ
ャー。



ハンガマン
言葉あてゲームは二人
の心を相手に伝えるの
に最速。遊びの中にユ
ーモアを持たせては。

APPLE II 取扱代理店 *印はローン及びクレジット可能

- 東京 コンピュータ・ラフ 1, 2 (03)812-4911
- アスター・インターナショナル (03)354-2661
- *西武百貨店サウンドハウス部 (03)981-0111
- ロジックハウス (03)363-2651
- *関ヶイフ (03)903-5551
- 大阪 コスモス新大阪 (06)305-5321
- 札幌 CQハードソン (011)821-1189

■代理店

コンピュータラブ1

〒113 東京都文京区本郷6-16-3
幸伸ビル2F ☎(03)812-4911

コンピュータラブ2

〒101 東京都千代田区神田佐久間町1-14
第2東ビル2F

使って 便利!

TK-80用

mini アセンブラ



中島 頼義

TK-80を使ってプログラムを作ろうとすると、アセンブラがないので(LKIT-16, H68/TRなどではある。)命令表を見ながら自分でアセンブルをしなければなりません。この作業は口で言うのはやさしいけれど、実際やると案外めんどくさなものです。特にMOV命令をマシンコードに変換するのは、いかに表をみればよいとはいえ……やっているうちに、目がチカチカしてきます。

そこで「命令コードをタイピングするとマシンコードを答えてくれるプログラムはできないものか」と考えて作ってみたところ、意外にも(?)TK-80上のメモリ(RAM1Kバイト)だけで、作れたので紹介します。「ハンドアセンブル」で苦勞している方は一度使ってみてください。

プログラム

Mini-Assemblerの概略フローチャートを図6に示します。KEY-11ルーチンとSEGCG+6ルーチンの2つは、ROMエリアにあるモニタールーチンを利用しています。命令コードを翻訳するためのテーブルとしてニモニク・テーブル(NMTBL)をもっています。このテーブルには図7に示すように各命令のタイプが登録されています。命令タイプとしては10種類あって、それを表3に示します。したがってマシンコードはこの10種類のタイプ別に求められることになります。

おわりに



このプログラムの一番の難点はLEDに表示する文字パターン(特にXとZ)でしょう。英文字を8セグメントのLEDで表示する事は最初からかなり無理があるのを承知で作ったとはいえにくいですが、改良の余地があると思われる方はアドレス8066~8076に表示用セグメント・データがあるので、そこを変えてもらえばよいと思います。メモリの都合ですべての命令がNMTBLに入りませんがメモリに余裕のある方はNMTBLを大きくして、アドレス8145の3Cを命令の個数に変えてください。たとえば64個ならば、40に変更します。

キーインする時に、いちいち図2を見てキーインするのは大変なので、私は図2のA~Zの文字を小さな紙に書いてキーボードに貼って使っています。このあたりはもっとよい工夫もできそうなので各自工夫してください。できましたら発表してください。

なお、プログラム中でACNTを1ずつ上げているのは次の機会にそれを表示するようにするためです。

必要な装置

TK-80 (RAM1Kバイト)(8000~83FF番地)

使い方

とっても簡単です。変換したい命令(ニモニク)コードをキーインするだけでLED上に16進数2桁でマシンコード(Operation Code)が表われてきます。もし、キーインした命令が誤っていたら、エラーとして「11」が表示されるので再び、キーインします。(図1、図2、図3参照)

アセンブルできる命令は表1に、またレジスタの指定は表2に示します。なお、キーインしたニモニク・コードはLED表示部の左6桁に表示されます。その時の各文字の表示パターンを図4に示します。

命令コードのキーイン方法は、図5の例に示すように、命令コードをLED上で先頭4桁に、レジスタ番号は第5、第6桁目にセットします。(図3参照)

図1 使い方

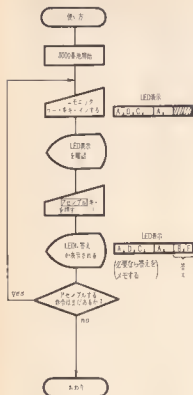


図5 操作例

- ① ADD A 命令は ?
- ② MOV A, M 命令は ?
- ③ PUSH BC 命令は ?
- ④ RLC 命令は ?
- ⑤ レジスタを必要とする命令は先頭3バイトのみキーインするだけでもよい。
- ⑥ HLT 命令は ?
- ⑦ RST 3 の命令は ?
- ⑧ STAX BC 命令は ?
- ⑨ STA 命令は ?
- あるいは

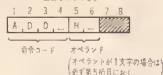
図2 キー配置と答の表示

LED表示部				
命令コード	レジスタ	データ	(16進数)	
X	Z			
S	T	U	V	
M	n	O	P	r
E	h	i	J	L
A	B	C	D	0

⑤ RST 命令のオペランドである数字は A~J を押す
16進キーの 0~F に対応している

図3 命令のキーイン方法

- オペランドの必要な命令の場合



- オペランドの不要な命令の場合

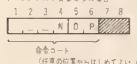


表1 アセンブル可能な命令の一覧表

1	ACI	21	JM	41	PUSH
2	ADC	22	JMP	42	RAL
3	ADD	23	JNC	43	RAR
4	ADI	24	JNZ	44	RET
5	ANA	25	JP	45	RLC
6	ANI	26	JPE	46	RRC
7	CALL	27	JPO	47	RST
8	CMA	28	JZ	48	SBB
9	CMC	29	LDA	49	SBI
10	CMP	30	LDA X	50	SHLD
11	CPI	31	LHLD	51	SPHL
12	DAA	32	LXI	52	STA
13	DAD	33	MOV	53	STAX
14	DCR	34	MOV	54	STC
15	DCX	35	NOI	55	SUB
16	HLT	36	ORA	56	SUI
17	IN	37	ORI	57	XCH
18	INR	38	OUT	58	XRA
19	INX	39	PCHL	59	XRI
20	JC	40	PDP	60	XTHL

図4 文字の表示パターン

123	4567
0 (00) 1 (01) 2 (02) 3 (03)	4 (04) 5 (05) 6 (06) 7 (07)
89AB	CDEH
8 (08) 9 (09) A (0A) B (0B)	C (0C) D (0D) E (0E) H (0F)

表2 レジスタの指定方法

レジスタは次の8種の内から選び
キーインする。

- ① A
- ② B
- ③ C
- ④ D
- ⑤ E
- ⑥ H
- ⑦ L
- ⑧ M

ペアレジスタは次の5種の内から選び
キーインする。

- ① BC
- ② DE
- ③ HL
- ④ AP ... TK80マニュアル上では、PSWと記されているが、ここではAPと表現する
- ⑤ SP

⑥ PUSH AP
LXI HL



図6 ミニ・アセンブラ・フローチャート

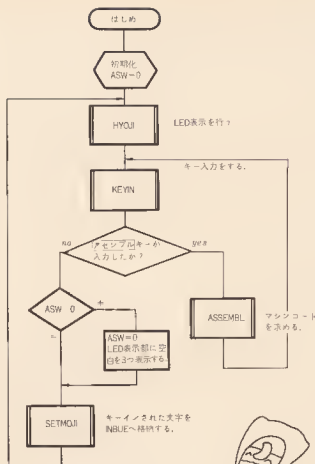


表3 命令タイプ

命令タイプ	オペランドの内容からなる マシンコードのビット番号	その値(3bit)	該当する命令
0	なし	——	NOP, ADI等
1	ba ~ ba, ba ~ ba	0 ~ 7	MOV
2	ba ~ ba	0 ~ 7	MVI, INR
3	ba ~ ba	0 ~ 7	DCR, ADD, SUB等
4	ba ~ ba	0, 2, 4, 6	LXI, INX
5	ba ~ ba	1, 3, 5, 7	PUSH, POP
6	ba ~ ba	0 or 2	DAD, DCX
7	ba ~ ba	0 or 3	STAX
8	ba ~ ba	0 ~ 7	LDAX
10	ニモニックを4桁で処理する必要のある命令	——	RST, LDA, LDAX, STAX, STA

左) ビット番号とは ba ba ba ba ba ba ba ba とする

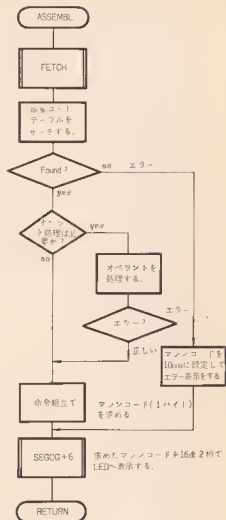
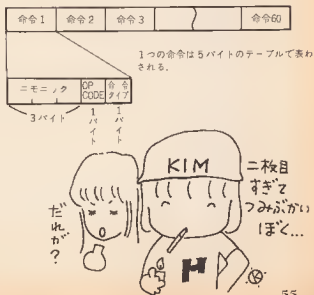


図7 ニモニック・テーブルの構成



~~~~~ 《ミニ・アセンブラ・プログラム・リスト》 ~~~~~

\*\*\*\*\*  
\* M I N - A S S E M B L E R \*  
\*\*\*\*\*

|               |                         |                                                   |                 |
|---------------|-------------------------|---------------------------------------------------|-----------------|
| 8000 CD 41 80 | MINIASS                 | CALL INITIAL                                      |                 |
| 8003 CD 90 80 | AA                      | CALL HYDJI                                        |                 |
|               | BB                      | EQU *                                             |                 |
| 8006 CD 16 02 |                         | CALL KEYIN                                        | A=DATA          |
| 8009 FE 12    |                         | CPI "12"                                          |                 |
| 800B FA 19 80 |                         | JM DD                                             | A=0, 1, ... 11  |
| 800E CA 06 80 |                         | JZ BB                                             | A=12 (IGNORE)   |
| 8011 FE 16    |                         | CPI "16"                                          |                 |
| 8013 CA E7 80 |                         | JZ ASSEMBL                                        | A="16" ASSEMBLE |
| 8016 F2 06 80 |                         | JP BB                                             | A="17" (IGNORE) |
| 8019 47       | DD                      | MOV B, A                                          |                 |
| 801A 3A 5A 80 |                         | LDA ASW                                           | ASSEMBL SWITCH  |
| 801D A7       |                         | ANA A                                             |                 |
| 801E CA 37 80 |                         | JZ FF                                             | ASW=0--> FF     |
|               | * OFF ASW               |                                                   |                 |
| 8021 AF       |                         | XRA A                                             | A=0             |
| 8022 32 FE 83 |                         | STA DIG7                                          |                 |
| 8025 32 FF 83 |                         | STA DIG6                                          |                 |
| 8028 32 5A 80 |                         | STA ASW                                           | ASW=0           |
| 802B 0E 03    |                         | MVI C, 3                                          |                 |
| 802D 3E 1F    |                         | MVI A, "1F"                                       |                 |
|               | OFFA1                   | EQU *                                             |                 |
| 802F CD D7 80 |                         | CALL SETL6                                        |                 |
| 8032 0D       |                         | DCR C                                             |                 |
| 8033 C2 2D 80 |                         | JNZ OFFA1                                         |                 |
| 8036 0D       |                         | NOP                                               |                 |
| 8037 CD B9 80 | FF                      | CALL SETMDJI                                      |                 |
| 803A C3 03 80 |                         | JMP AA                                            |                 |
| 803D 0D 0D 00 |                         | DC 0, 0, 0                                        |                 |
| 8040 0D       |                         | NDP                                               |                 |
|               | INITIAL                 | EQU *                                             |                 |
| 8041 3E 01    |                         | MVI A, 1                                          |                 |
| 8043 32 5A 80 |                         | STA ASW                                           | ASW=1           |
| 8046 21 0D 00 |                         | LXI HL, "0000"                                    |                 |
| 8049 22 5B 80 |                         | SHLD ACNT                                         | ACNT=0          |
| 804C 22 FE 83 |                         | SHLD DIG7                                         |                 |
| 804F C9       |                         | RET                                               |                 |
| 8050 0A       | L1                      | DC "0A"                                           | A               |
| 8051 16       | L2                      | DC "16"                                           | S               |
| 8052 0E       | L3                      | DC "0E"                                           | E               |
| 8053 12       | L4                      | DC "12"                                           | M               |
| 8054 0B       | L5                      | DC "0B"                                           | B               |
| 8055 1D       | L6                      | DC "1D"                                           | L               |
| 8056 1F       | L7                      | DC "1F"                                           |                 |
| 8057 1F       | L8                      | DC "1F"                                           |                 |
| 8058 0D       | ACNT                    | DC 0, 0                                           |                 |
| 805A 0D       | ASW                     | DC 0                                              |                 |
| 805B 0D       | DPCODE                  | DC 0                                              |                 |
| 805C 0D       | DPTYP                   | DC 0                                              |                 |
| 805D 0D 0D 00 | NCDD1                   | DC 0, 0, 0, 0, 0, 0                               |                 |
| 805E 0D 0D 00 |                         |                                                   |                 |
|               | NCDD4                   | EQU NCDD1+3                                       |                 |
|               | NCDD5                   | EQU NCDD1+4                                       |                 |
|               | NCDD6                   | EQU NCDD1+5                                       |                 |
|               |                         | DC 0, 0, 0                                        | **YDBI**        |
| 8063 0D 0D 00 | SEGTBL                  | DC "74", "04", "0D"                               | H I J           |
| 8066 74 04 0D |                         | DC "37", "54", "3F"                               | M N O           |
| 8069 37 54 3F |                         | DC "73", "6D", "07"                               | P S T           |
| 806C 73 6D 07 |                         | DC "3E", "1C", "03"                               | U V Z           |
| 806F 3E 1C 03 |                         | DC "E4", "0D", "3E"                               | X - L           |
| 8072 E4 0D 3E |                         | DC "7D", "0D", "FF"                               | R SP-           |
| 8075 7D 0D FF |                         | DC "FF"                                           | -               |
| 8078 FF       |                         | NDP                                               |                 |
| 8079 0D       |                         |                                                   |                 |
| 807A 0B 0C    | RBTBL                   | DC "0B", "0C", "0D", "0E", "0F", "1D", "12", "0A" |                 |
| 807C 0D 0E    | *                       | B C D E H L M A                                   |                 |
| 807E 0F 1D    | * NAME OF REGISTER      |                                                   |                 |
| 8080 12 0A    |                         |                                                   |                 |
| 8082 0B 0C    | PRTEL                   | DC "0B0C", "0D0E", "0F1D", "1615", "0A15"         |                 |
| 8084 0D 0E    | *                       | BC DE HL SP AP                                    |                 |
| 8086 0F 1D    | * NAME OF PAIR-REGISTER |                                                   |                 |



```

8088 16 15
808A 0A 15
808C 00 00 00
808F 00

HYOJ: EQU *
      LXI HL, "8050"
      LXI DE, "83F8"
      MVI C, 6

*
HY1 MOV A, M
    CALL CNVSEG
    STAX DE
    INX HL
    INX DE
    DCR C
    JNZ HY1
    RET

CNVSEG EQU *
      PUSH BC
      PUSH HL
      LXI HL, SEGO
      CPI "0F"
      JM CNV1
      LXI HL, SEGTBL-15
      MOV C, A
      MVI B, 0
      DAD BC
      MOV A, M
      POP HL
      POP BC
      RET

CNV1 MOV C, A
      MVI B, 0
      DAD BC
      MOV A, M
      POP HL
      POP BC
      RET

SETMOJ: EQU *
      LXI HL, <L3>
      LXI DE, <RST>
      MVI C, 4
      CALL COM8YT
      MOV A, B
      JNZ SET30
      ANI 07
      JMP SET40
      ADI 10
      JMP SETL6
      RET

SET30 ADI 10
      JMP SETL6

SETL6 EQU *
      PUSH BC
      LXI HL, <L6>
      MVI C, 6
      MOV B, M
      MOV M, A
      MOV A, B
      ODX HL
      DCR C
      JNZ SETL610
      PDP BC
      RET

SETL610 MOV B, M
      MOV M, A
      ODX HL
      DCR C
      JNZ SETL610
      PDP BC
      RET

*
RST DC "1E16", "171F" *RST*

*
JMP ASSEMBL
* ++++++
* JMP BB (END)
ASSEMBL EQU *
      CALL FETCH
      CALL LOOKUP
      CALL MAKEOP
      CPI "FF"
      JNZ ASBA
      MVI A, "10"
      STA OPCODE
      XRA A
      EQU *

80E7 C0 19 81
80EA CD 41 81

*
80ED CD 93 81
80F0 FE FF
80F2 C2 FB 80
80F5 3E 10
80F7 32 58 80
80FA AF

ASBA EQU *

```

DC 0.0. /

\* YOB1 \*

DISPLAY ROUTINE

CONVERT TO SEG-DATA (LED)

CONVERT ROUTINE

A=00-0E

GET SEGMENT-DATA

SET MOJ1 IN L6

B= (HL)  
(HL) = A  
A = B

FETCH MNEMONIC CODE

LOOK UP NMTBL

MAKE OPERAND

ERROR?

NOT

OPCODE = "10"

A = 0



```

80FB 21 58 80      LXI  HL, "CODE
80FE 06             ORA  M
80FF 77             MOV  M, A
8100 21 F7 83       LXI  HL, DISP4
8103 77             MOV  M, A
8104 11 FE 83       LXI  DE, DIG7
8107 C0 C6 01       CALL SEGCG+6
810A 3E 01          MVI  A, "01"
810C 32 5A 80       STA  ASW
810F 2A 58 80       LHLO ACNT
8112 23             INX  HL
8113 22 58 80       SHLO ACNT
8116 C3 05 80       JMP  BB
                        // END OF ASSEMBLE

8119 21 50 80       *
811C 0E 06          LXI  HL, 'NCOD1'
811E 3E 1F          MVI  C, 6
                        A=SPACE

8120 77             *
8121 23             FE141L MOV  M, A
8122 0D             INX  HL
8123 C2 20 81       DCR  C
                        CLEAR WITH SPACE
                        JNZ  FE141L

8126 21 50 80       *
8129 0E 06          LXI  HL, 'L1'
812B 8E             MVI  C, 6
812C C2 35 81       FE09L CMP  M
812F 23             JNZ  FE10A
8130 0D             INX  HL
8131 C2 28 81       DCR  C
8134 C9             JNZ  FE09L
                        SPACE?
                        NOT

8135 11 50 80       FE10A EQU  *
                        LXI  DE, 'NCOD1'
                        ERROR (ALL SPACES)

8138 7E             *
8139 12             FE12L MOV  A, M
813A 23             STAX DE
813B 13             INX  HL
813C 0D             INX  DE
813D 00             DCR  C
813E C2 38 81       JNZ  FE12L
8140 C9             RET

*
* CALL LOOKUP
* A = OP-TYP
LOOKUP EQU *
LXI HL, 'NMTBL
MVI B, 60
LXI DE, 'NCOD1'

8141 21 70 82       *
8144 06 3C          MVI  B, 60
8146 11 50 80       LXI  DE, 'NCOD1'

8149 0E 03          *
814B CD 83 81       LOOK11L MVI  C, 3
814E CA 60 81       CALL  COMBYT
8151 23             JZ   LOOK12A
8152 23             INX  HL
8153 23             INX  HL
8154 23             INX  HL
8155 23             INX  HL
8156 05             INX  HL
8157 C2 49 81       OGR  B
                        FOUND EXIT
                        +1
                        +2
                        +3
                        +4
                        +5

815A 3E FF          ** ERROR (NOT FOUND)
815C 32 5C 80       MVI  A, "FF"
815F C9             STA  OPTYP
                        SET ERROR FLAG
                        RET

*
* /FOUND//
LOOK12A INX  HL
INX  HL
INX  HL
INX  HL
MOV  A, M
CPI  Y0
JNZ  LOOK12B
LOA  NCOD4
DP-TYP=10?

```



60 OPERATION

COMPARE 3 BYTES

FOUND EXIT

+1

+2

+3

+4

+5

SET ERROR FLAG

DP-TYP=10?

るのはI/Oしかありません。まあ、それからM6800ファンの方に面白いおはなしを… M6800は8Di<sub>8</sub> (9Di<sub>8</sub>かもしれない) ガサツ (I/Oがさびしてる者) あ、やっぱり9DだわBDはBSRじゃ、とにかく9Dを実行するとコンピュータは暴走するのです。IRQはもちろんNM Iもききません、ついでにRESもきかなくてPOWERを切らなきゃとまらないようにすればいいのに…) RESをかけるしか俺に手はないのです、という話を聞いた覚えがあります。日立のほうはどうかかわりません、実験をするアホがいらっしゃらなければレポートをI/O元気

```

816D FE 16          CPI    "16"          Y XT 1
816F C2 77 81       JNZ    LOOK12B      NOT
8172 23             INX    HL            +5
8173 23             INX    HL            +1
8174 23             INX    HL            +2
8175 23             INX    HL            +3
8176 23             INX    HL            +4

*
8177 7E             MOV    A, M          GET OPIYP
8178 32 5C 80       STA    OPTYP
8178 57             MOV    D, A

*
817C 28             DCX    HL            +3
817D 7E             MOV    A, M          GET OPCDDE
817E 32 58 80       STA    OPCODE

*
8181 7A             MOY    A, O          A=OPTYP
8182 C9             RET

COMPSY1 EQU *          COMPARE N BYTE
      PUSH HL
      PUSH DE
      LDAX OE
      CMP M            (DE) > (HL) 0000
      JNZ CDMP90       (DE) = (HL) ZERO ON
      INX HL           (DE) < (HL) CARRY ON
      INX OE
      DCR C
      JNZ COMP10L
      EQU *
      POP DE
      POP HL
      RET

*
* CALL MAKEDP
* OUT A = OP-TYPE (A="FF" IF ERROR)
*
MAKEDP EQU *
      LDA OPTYP
      CPI "FF"        LOOK UP ERROR?
      RZ              YES
      CPI 01          TYP 0
      JM ASB0
      JZ TYP1
      CPI 03          TYP 1
      JM TYP2
      JZ TYP3
      CPI 05          TYP 2
      JM TYP4
      JZ TYP5
      CPI 08          TYP 3
      JM TYP6
      JZ TYP7
      CPI 10          TYP 4
      JM TYP8
      JZ TYP9
      CPI 10          TYP 5
      JM TYP10
      JZ ASB0
      JZ ASB0          TYP 6 & TYP 7
      JZ ASB0          TYP 8
      JZ ASB0          TYP 9
      JZ ASB0          TYP 10

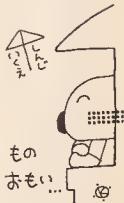
* ERROR
      MVI A, "FF"
      RET

*
* TYP 0 (TYP 10)
*
ASB0 EQU *
      XRA A            A=0
      RET

*
* TYP 1
*
TYP1 EQU *
      CALL TYP2
      MOV B, A

*
      LDA NC006
      CALL LOOKR
      ORA 8
      GE1 SSS
      A=0000DSSS

```



女子のための実験室までお寄せください。☆これにて小生のいいたいことは終りと思つたら大丈夫がい。それでは最後に2言3言、1E18の意味おしえてくれ。/M6800エライノ1/0エライノここまでよんでくださつてどうもありがとうございます。これで心おきなく死ぬことが...いやいやいや死ぬのはいややじゃーノそれではサヨウナラ。

(東京都 乙女の横山秀樹)



```

81CE C9          RET
                  EQU *
81CF CD 06 81    TYP2      CALL TYP3
81D2 07          TYP2RLC3  RLC
81D3 07          RLC
81D4 07          RLC
81D5 C9          RET

*
*
*
TYP3            EQU *
81D6 3A 61 80    LDA      NCDD5
81D9 00          NDP

*
*
LDDKR           EQU *
81DA E5          PUSH HL
81DB C5          PUSH BC
81DC 21 7A 80    LXI HL, (RGTBL)
81DF 0E 08       MVI C, 8
81E1 BE          LOOKR16L  CMP M
81E2 CA EF 81    JZ       LDDKR16A
81E5 23          INX HL
81E6 00          DCR C
81E7 C2 E1 81    JNZ      LDDKR16L

* ERRDR
81EA 3E FF       MVI A, "FF"
81EC C3 F2 81    JMP      LOOKR16B
81EF 3E 08       MVI A, 8
81F1 91          SUB C
81F2 C1          EQU *
81F3 E1          POP BC
81F4 C9          POP HL
                  RET

*
* TYP4            (8C, DE, HL, SP, DR, AP)
*
TYP4            EQU *
81F5 DE 05       MVI C, 5
81F7 00          NDP

* LDDK UP PAIR REGISTER
*
LDDKPR          EQU *
81F8 E5          PUSH HL
81F9 D5          PUSH DE
81FA 06 00       MVI B, 0
81FC 21 82 80    LXI HL, (PRTBL)
81FF 11 61 80    LXI DE, NCDD5

*
LOOKP18L        LDAX DE
8203 8E          CMP M
8204 23          INX HL
8205 C2 DF 82    JNZ      LDDKPI8B
8208 13          INX DE
8209 1A          LDAX DE
820A 1B          DCX DE
820B 8E          CMP M
820C CA 1A 82    JZ       LOOKP18C

*
LDDKPI8B        INX HL
820F 23          INR B
8210 04          DCR C
8211 00          JNZ      LOOKP18L
8212 C2 02 82    JNZ      LOOKP18L

* ERRDR (NOT FOUND)
8215 3E FF       MVI A, "FF"
8217 C3 27 82    JMP      LOOKPI9E

*
* FOUND
*
LDDKPI8C        EQU *
821A 78          MOV A, 8
821B FE 04       CPI 04
821D C2 21 82    JNZ      LDDKPI8D

```

TYP3

EQU \*  
LDA NCDD5  
NDPEQU \*  
PUSH HL  
PUSH BC  
LXI HL, (RGTBL)  
MVI C, 8  
LOOKR16L CMP M  
JZ LDDKR16A  
INX HL  
DCR C  
JNZ LDDKR16L\* ERRDR  
MVI A, "FF"  
JMP LOOKR16B  
LDDKR16A MVI A, 8  
SUB C  
LDDKR16B EQU \*  
POP BC  
POP HL  
RET\* TYP4 (8C, DE, HL, SP, DR, AP)  
\*  
TYP4 EQU \*  
MVI C, 5  
NDP

\* LDDK UP PAIR REGISTER

\*  
LDDKPR EQU \*  
PUSH HL  
PUSH DE  
MVI B, 0  
LXI HL, (PRTBL)  
LXI DE, NCDD5\*  
LOOKP18L LDAX DE  
CMP M  
INX HL  
JNZ LDDKPI8B  
INX DE  
LDAX DE  
DCX DE  
CMP M  
JZ LOOKP18C  
A=NCDD5  
NOT-(1ST BYTE)  
A=NCDD6  
FOUND (OK)\*  
LDDKPI8B INX HL  
INR B  
DCR C  
JNZ LOOKP18L  
\* ERRDR (NOT FOUND)  
MVI A, "FF"  
JMP LOOKPI9E  
B=B+1  
C=C-1\*  
\* FOUND\*  
LDDKPI8C EQU \*  
MOV A, 8  
CPI 04  
JNZ LDDKPI8D  
A=4  
NDP

74+~

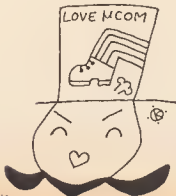


[1E]はコマンド・テーブルのエンドコードとして使われています。たとえば、\$0122番地から52、55、4E、1E、08、76、4C、49……と入っていますね。これはR、U、N、1E、0876、L I……つまりRUNというASCIIコードと、インポートされたコマンドとを比較するテーブル・データの最後を意味します。1Eの後の2バイト・データがそのコマンドを処理する番地を示しています。この場合、RUNというコマンドは\$0876番地から処理ルーチンがはじまります。—編—]

```

8220 3D                                DCR    A                    A-3
*
* LOOKP18D    RLC                0000 0100
8221 07                                RLC                0000 1100
8222 07                                RLC                0001 1000
8223 07                                RLC                0011 0000
8224 07                                ANI    "3B"            70XX X000
8225 E6 3B
*
* LOOKP19E    EQU    *
8227 01                                POP    DE
8228 E1                                PDP    HL
8229 C9                                RET
*
* TYP 5                (BC, DE, HI OR SP)
*                    1 3 5 7
TYP5    EQU    *
822A 0E 04                MVI    C, 4
822C CD F8 B1            CALL    LOOKPR
822F FE FF                CPI    "FF"
8231 C8                    RZ
8232 C6 08                ADI    8
8234 C9                    RET
*
* TYP6 & TYP7
*                    (BC OR DE)
*
TYP67    EQU    *
8235 57                MOV    D, A                    D=6OR7 (OPTYP)
8236 0E 02                MVI    C, 2
8238 CD F8 B1            CALL    LOOKPR
8238 FE FF                CPI    "FF"
823D C8                    RZ
*
823E 4F                MOV    C, A
823F 7A                MOV    A, D                    GET OPTYP
8240 FE 07                CPI    07                    OPTYP=7T
8242 79                MOV    A, C
8243 C0                    RNZ                    / TYP6 /
8244 C6 08                ADI    08                    / TYP7 //
8246 C9                    RET
*
* TYP8 (RST0 RST7)
*
TYP8    EQU    *
8247 3A E1 B0            LDA    NCOD5
824A 07                                RLC                0000 XXX0
8248 07                                RLC                000X XX00
824C 07                                RLC                00XX X000
824D C9                    RET
*
* MNEMONIC CODE TABLE
*                    1 ELM SIZE = 5 BYTES
*
NMTRL    EQU    *
8270 0A 0C 10 CE 00            DC    "0A0C10", "CE", 0 1 ACI
8275 0A 0D 0C 88 03            2 ADG
827A 0A 0D 0D 80 03            3 ADD
827F 0A 0D 10 C6 00            4 ADI
8284 0A 13 0A A0 03            5 ANA
8289 0A 13 10 E6 00            6 ANI
828E 0C 0A 10 CD 00            7 CAL
8293 0C T2 0A 2F 00            8 CMA
8298 0C 12 0C 3F 00            9 CMC
829D 0C 12 15 B8 03            10 CMP
82A2 0C 15 10 FE 00            11 CPI
82A7 0D 0A 0A 27 00            12 DAA
82AC 0D 0A 0D 01 05            13 DAD
82B1 0D 0C 1E 05 02            14 DCR
82B6 0D 0C 1B 03 05            15 DCX
82B8 0F 1D 17 76 00            16 HLT
82C0 10 T3 1E 04 02            17 INR
82C5 10 13 1F D8 00            18 IN
82CA 10 13 1B 03 04            19 INX

```



もてない  
君にごめいふくを...

82C1 11 11 1F DA 00  
 82D4 11 12 1F FA 00  
 82D9 11 12 15 C3 00  
 82DE 11 13 0C D2 00  
 82E3 11 13 1A C2 00  
 82E8 11 15 1F F2 00  
 82ED 11 15 0E EA 00  
 82F2 11 15 14 E2 00  
 82F7 11 1A 1F CA 00  
 82FC 1D 0D 0A 3A 0A  
 8301 1D 0A 18 02 07  
 8306 1D 0F 1D 2A 00  
 830B 1D 18 10 01 04  
 8310 12 14 19 4D 01  
 8315 12 19 10 06 02  
 831A 13 14 15 0D 00  
 831F 14 1E 0A 8D 03  
 8324 14 1E 10 F6 00  
 8329 14 18 17 03 00  
 832E 15 0C 0F E9 00  
 8333 15 14 15 C1 04  
 8338 15 18 16 C5 04  
 833D 1E 0A 1D 17 00  
 8342 1E 0A 1E 1F 00  
 8347 1E 0E 17 C9 00  
 834C 1E 1D 0C 07 00  
 8351 1E 1E 0C 0F 00  
 8356 1E 16 17 C7 08  
 835B 16 0B 08 98 03  
 8360 16 0B 10 DE 00  
 8365 16 0F 1D 22 00  
 836A 16 15 0F F9 00  
 836F 16 17 0A 32 0A  
 8374 16 0A 18 02 06  
 8379 16 17 0C 37 00  
 837E 16 18 0B 9D 00  
 8383 16 18 10 D6 00  
 8388 1B 0C 0F E8 00  
 838D 1B 1E 0A A8 03  
 8392 1B 1E 10 EE 00  
 8397 1B 17 0F E3 00  
 839C 0D 00 00 00 00  
 83A1 0D 00 00 00 00  
 83A6 0D 00 00 00 00  
 83AB 0D 00 00 00 00



20  
 21 JM  
 22 JMP  
 23 JNC  
 24 JNZ  
 25 JP  
 26 JPE  
 27 JPO  
 28 JZ  
 29 LDA  
 30 LDAX  
 31 LHLD  
 32 LXI  
 33 MOV  
 34 MVI  
 35 NOP  
 36 ORA  
 37 ORI  
 38 OUT  
 39 PCHL  
 40 PDP  
 41 PUSH  
 42 RAL  
 43 RAR  
 44 RET  
 45 RLC  
 46 RRC  
 47 RST  
 48 SBB  
 49 SBI  
 50 SHLD  
 51 SPHL  
 52 STA  
 53 STAX  
 54 STC  
 55 SUB  
 56 SUI  
 57 XCHG  
 58 XRA  
 59 XRI  
 60 XTHL  
 61 予 備  
 62 予 備  
 63 予 備  
 64 予 備

## New Products

### § マイコン・ラボラトリーの開設 §

■ASR (オートメーション システム リサーチ)  
 では、システムハウスとユーザーが一緒になってマイコンの応用システムを開発するため、新橋マイコン・ラボラトリーを開設した。

同社では幅広い応用分野が見込まれているマイコンも、ユーザーが応用システムを開発するには、まだ専門的知識や経験、設備が不足している現状を打開するため、ユーザーの立場になった共同開発、コンサルティング、教育などの援助を行なうとしている。

このラボには、米・DEC社のPDP-11/34をはじめ、NOVA、HP-21MXなどのミニコンを設置しており、近くインテル社のMDSなども設置される予定。

ASR社では、DEC社の16ビット・マイコンを、主力製品として、今後、マイコンの新規ユーザーの開拓に期待を寄せ、その一環として、マイコンの応用システムの開発、コンサルティング業務だけでなく、開

発ツールの貸出し、定期的なセミナーの開催なども積極的に進めようとしている。

| 項 目              | 単 位            | 料 金           | 項 目                                     |
|------------------|----------------|---------------|-----------------------------------------|
| ASR-2025ライブラリ使用料 | 30分            | 500円          | オンライン/オフライン                             |
| 1602A 目 録        | 1冊             | 4,500円        | リストより                                   |
|                  | 1冊             | 1,500円        | 標準テープより                                 |
| プログラムの印刷料        | 1時間            | 5,000円        | マイクロコンピュータ<br>ミニコンピュータ (NOVA, PDP-21MX) |
| 計算機オペレーション料      | 1時間            | 5,000円        | エディタ、イメージ、アセンブリ                         |
| 各種クロスアサリ料        | 30分            | 1,500円        | 8080, 8088, 286, その他                    |
| PORT 780         | 30分            | 1,500円        | MP-88, 286, 386, 486                    |
| プログラムの印刷料        | 1冊             | 5,000円        | NOVAのみ使用可能                              |
| テープ              | 計算機利用<br>1時間以上 | 30分1巻以上は別料金   |                                         |
| L.P. 用紙          |                | 30分250円以上は別料金 |                                         |
| 印刷料 (30分) 使用料    | 30分            | 1,500円        | 20分 1,100円                              |

サービス時間 月 00 - 17:30

〈問い合わせ先〉オートメーション システム リサーチ ㈱  
 105 東京都港区西新橋3-5-8 西新橋中央ビル  
 ☎ (03) 437-5471



RANDOM  
Box

## TK-80の割り込み

## 前略 TK-80 マン様

I/O 3月号であなたの投書を見ましたので、ひとつの解決法を考えてみたいと思います。私は現在、Z-80 CPUで自作中ですが、TK-80については、前に 度々マニュアルなどをとりよせて検討したことがあります。

しかし実際に、TK-80をもっているわけではないし、実験結果でもないで、一応参考にして下さい。完全に制作する自信はあるのですが、...

## TK-80の改造

TK-80は、8228を特殊モードで使っているので、このままではRST 0-RST 6は使えません。8228のINTA端子を+12Vにプルアップすると、8228はINT信号への対応として、つねにRST 7を出力するような特殊モードになる

のです。

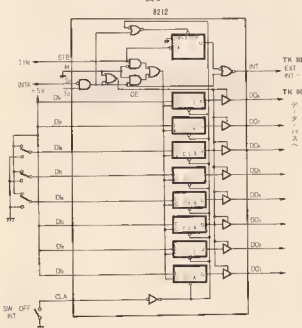
割り込みには、必ずINTA信号(interrupt acknowledge)が必要ですから、INTA端子をスイッチでプルアップ抵抗と、INTA出力とに切りかえられるようにしてはなりません。回路図は図1のとおりです。

ご存知でしょうか、TK-80ではRST 7を使ってSTEP動作をさせていますから、特殊モードを完全に切りはなすのは、いきさしんどいわけです。ですからSWで切り替にしました。それからSTEP動作にスイッチを倒しておくこと、INTA信号がでますから、必ずAUTO側に倒しておくことも必要です。

## 割り込み回路(図2)

これは、割り込み信号を発生する部分と、INTA信号に对应してBUSに信号をのせる部分に分れます。ここでは最も簡単なということで8212 1個だけで構成してみました。8212は、データバス(8bit) 3 state buffer(8bit) それに割り込み処理用の回路まで1つのICになっているうえに、安いという強みがありますからね。(見かけ上では、富士通の8212コンパチのIC MB471が¥540で入ります。)

図2



きます。)

8212は、INPUT PORTのモードで使います(Mをアース)。データ入力は、RST 1-RST 6の命令コードとしてスイッチで設定するようにしてありますが、ほかの命令、たとえば PCHL のようなジャンプ命令などゲームには便利ではないでしょうが、うまく使おう。レジスタ・マシンの 8080 のことですから、Relative Jumpでも何でもこなせるでしょう。割り込みは、RST 命令だけなんだからいいこと考えるくてもいいのではないでしょうが、まあ、データ入力はジャンプが何かにして、別の命令をのせてみるのもおもしろいと思います。

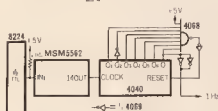
データ出力はもちろんデータバスへです。INTA信号はSTBの立ち上がりで出ます。STBには1Hzのハルスを入れますと、当然INTAも1秒ごとに立ちますね。OUTPUT enableは、S1とS2とANDですが、ここではS1をINTAにつないでS2はOpenのままです。これでINTAが入ると、OUTPUT enableとなって、データバスに命令がのります。また同時に、INTAのF/Fのクリアにもなっているわけです。(うまくできているでしょう)ただし、この8212の使い方は少々変則的で、ふつうの使い方はちがいます。

最後にクリアのSWですが、これはゲームを始める前や、不要のときINTA信号がでないようにするためのものです。INTAは、TK-80のEXT INTに入れます。(この端子は真理論入力ですからINTAをそのまま入れていいのです。)

## 1秒ごとのパルス

ゲーム用のいいかげんなものでもいいのなら、555で発振器を作るのがいいでしょう。(うまく調整すればかなり精度は上がります。)

図3



う少し遅れる...というときは、商用周波数を分周するのがいいでしょう。しかし電流が自作でないときには、ちょっと商用周波数をとり出すのがめんどくさくなりますね。

正確でゆくなら、クロック・パルスを分周するのがいいでしょう。なせ、2048×10<sup>6</sup>分周、つまり1F 4000分周するのだから、TTLではちよっとばかり、しんどいでは、そこでC-MOSを使ってみました。(図3) クロック・パルスは8224の6ピンの出力(TTL)があそんでいるので、ここから取ります。初段のMSM5562はOKIの特許利用クロックのためのICです。これはC-MOSでありながら3MHz以上でも分周できるのでこれにしました。

ふつうのC-MOSでは最大2MHz (type) ということですから、2,048MHzで完結するかどうかかわからないからです。(ICのばらつきによって1MHzをそこそこしか動かないものもありますから。) もしもふつうのC-MOSが使えるなら、4020のほうが安いです。次段の4040は、4068で検出して強制リセット式の125分周にしています。出力で4068を並列にしているのは、Bufferです。

以上、最も簡単と思われる1秒ごとの割り込み回路を書きましたが、何か疑問があったら、月のおよぶ範囲で答えたいと思います。こちらもまだ勉強中の毎ですので、期待はしないでください。乱文にて失礼。今後ともよろしく。

(読者服 胡崎陽一)

図1



# 盲点におちこんだプログラムを 泥沼から救う

HELP

トレーサ

TRACER (Version 3.1)

玉村卓也



「プログラムは思った通りには動かない、命令語のとおり動く」とは、プログラマーの金言集のトップに書かれている格言です。複雑なフローを折って量んで極力圧縮し、テクニックを講じて、これでよい筈だと思って走らせてみると、どうもうまく働かない、人間は一度誤った確信をもっと、虫を発見する力が落ちてしまつてみえて、ドロ沼に入るとアガくほど深みに入つてわけがわからなくなり、ついには「CPUをこわしてしまつたのか?」と疑いだす始末とまいります。私自身の恥をお話すると、SU1と書くべきところをSB1と書いてしまい、この虫が長いこと発見できませんでした。

## トレーサとは

トレーサは、プログラムの1ステップごとにレジスタの内容や、スタック・ポインタ、フラグなどをプリントするもので、いうなれば、HardwareでOne Step動作をさせて、その際、1つ1つレジスタの内容をキーインによって呼び出し、それを記録する仕事を自動化してプリント・アウトさせるものです。

この目的と似たものにモニターROMを変更して、TVにDisplayさせるCRT Debuggerが、商品として発売されていますが、このトレーサはハードコピーこれを得ようとするもので、割り込みを使っていないので、DI命令があってもこれに関係なく<sup>※1</sup>内容が表示され、得た結果を机上でじっくりとたどってみると、ハタと手を打って「嘘吾愚者哉」と目がさめるわけです。

トレーサというDebug手段は、ふた昔前のコンピュータにも存在し、CRT Debuggerなどという結構な手段はまったく考えられなかった昔では、Memory Dumpで

は見当がつかず、One Step動作でもつかめない虫には、トレーサが特効薬だったわけです。副作用としてプリント用紙を湯水のように使うことと、貴重なコンピュータを長時間占有するという2大欠点があったわけで、今日のマイコンでは、有効稼働率が問題にならなくなったとはいえ、高価な放電プリンタのロールをバカスカ消費するのは少なからず気がひけるのは事実です。放電プリンタのロールは裏紙を使うわけにもいかず、水資源のとほしい日本では、トレーサはやはり奥の奥の手であるわけです。しかし、トレーサによって、命令語の解説書の片隅にふれてあっても念頭に述していなかったような、Hardwareの性質が自らのものにさらされるわけで、トレーサの家庭教師としての役割は捨てがたいものがあります。

## トレーサの機能

このトレーサには、逆アセンブラの機能が組み込まれています。何しろ8080の命令語は244種にもおよび、機械語だけの表現では私ごとく粗製顔には実用になえないので、しよせんレジスタの表現が32桁でおさまらず、2行のプリントになるため、TK-80様にMnemonic Decodeの行数もあわせてお願いしています。Print Out Formatは図1のとおりです。

このうちMの部分は、

|         |         |         |
|---------|---------|---------|
| ADD R,M | ADC R,M | SUB R,M |
| SBB R,M | ANA R,M | XRA R,M |
| ORA R,M | CMP R,M | INR M   |
| DCR M   |         |         |

の10種の命令の場合に限り表示されます。また、フラグの部分はサイン (+または-) パリティの奇偶 (O

※1. DIがあると割り込みを使ったOne Step動作はきかなくなります。

※2. SPHLが一方通行でなく、XTHLのように交換命令だったらフラグの処理に苦勞しなくてすむのですが。

## 図1 TRACER PRINT FORMAT

第1行目

| アドレス |   |   |   | マシンコード |   |   |   | OP |   |   |   | ニモニツク・オペラン |   |   |   | フラグ |   |   |   |
|------|---|---|---|--------|---|---|---|----|---|---|---|------------|---|---|---|-----|---|---|---|
| X    | X | X | X | X      | X | X | X | X  | X | X | X | X          | X | X | X | X   | X | X | X |

第2行目

| SP |   |   |   | TOP OF STACK |   |   |   | M |   | A |   | BC |   |   |   | DE |   |   |   | HL |   |   |  |
|----|---|---|---|--------------|---|---|---|---|---|---|---|----|---|---|---|----|---|---|---|----|---|---|--|
| x  | x | x | x | x            | x | x | x | x | x | x | x | x  | x | x | x | x  | x | x | x | x  | x | x |  |

つフラグ以降はすべて、この命令を実行したあとの各々の内容をあらわす。

□Mについては、本文参照。

またはE)は必ず表示され、ゼロ(Z)、減算(S)、C Y 4(Y)、キャリー(C)は、それらのビットが立ったときに表示されます。

トレーサもアセンブラ同様、I Qの高いトレーサもあれば鈍すものもあります。I Qの高いトレーサであるためには、命令語をどのような変った使い方をしても、完全にこれが追跡できなければ困るわけです。むしろそういった常識的でない使い方をした命令語が、トレースできなければ、トレーサの意味は激減してしまいます。一例をあげると、POPは必ずしもスタックからの拾い出しに使うとは限らないわけで、SPに必要な値を与えて表を読み出すのにも使えます。したがって、POPしたあと下側のスタックの内容は決して不要のものとはいきません。ところがここで割り込みがかかる、リターン・アドレスが書き込まれ、続いてT K-80モニタのブレーク・エントリでユーザー側のSPの値のままでPUSH PSWがでる(0151~0155)ので、メモリ内容が4バイト分書き換えられてしまいます。(これを防ぐためにはSPにテーブル・アドレスを与えているあいだDIをだしておくことが必要です)

このトレーサでは、DIに関係なく下側のスタックをSave Restoreしています。というのはSPを知ろうとすればDAD SPしか方法がなく、DAD SPをだすとフラグが変わるジャンルがあり<sup>12</sup>、やむを得ず\$9E8Cから\$9E9Eにかけて、その処理をしています。ただし、SPにROMエリアとかメモリ実装のないアドレスをあたえると、r AとFLAGは正しい結果が得られませんかので気をつけてください。

以上の注意だけでトレーサは万能かというに決してそうではなく、タイピングに関係するものには無力で、トレースしてみても全く意味をなしません。<sup>13</sup> したがって、I/Oサブルーチンなどは一括してトレースを除外する必要があります。また完全に動作することが確かな部分には、トレースする必要はありません。このトレーサは上記の目的のため、トレース・スキップ・

サブルーチン・テーブル(TSST)の登録と、部分トレースの2つの方法を用意しています。

TSSTの登録はさらに2種におかれます。まず、Lower Boundary (LB通常\$1000)未満、Upper Boundary (UB通常\$E0000)以上のエントリ・アドレスをもつサブルーチン(条件付Callを含む)をCallした場合トレースされません。次に、LB以上でUB未満のアドレスをエントリとするサブルーチンは、TSSTにエントリを個々に登録することによって、このサブルーチンをトレースさせないことができます。<sup>14</sup> またこの登録は、Runningの途中で追加することができますので、一応トレースしたサブルーチンは2回目からノー・トレースにすることができます。これらノー・トレース・モードは、あくまでCALL命令のエントリとして、上のいずれかの条件に入ったときのみ有効であり、それ以外の命令で、たとえばRST 7やJMP 0038などで領域内に入った場合はもちろんトレースされます。ノー・トレース・モードではCALL命令は1つのマクロ命令となり、プリントにはそのCallを完了した後のレジスタやフラグの状態が表示されます。

重要な注意として、トレース・スキップできるサブルーチンは、標準的な戻り方をするものに限られるのを忘れないでください。標準的な戻り方とは、そのサブルーチンをCallした次のアドレスに戻ることをいい、サブルーチン内でReturn Addressを加工して別のところへ戻るようなサブルーチンは、トレース・スキップさせることはできません。<sup>15</sup> もしこのようなものをTSSTに登録しますと、正しい動作が得られず暴走する。またはトレーサ側からいつまでもそのサブルーチンの実行中の状態となってトレーサに戻ってこないかのいずれかになります。登録されていなければ、そのアドレス加工を含めて、内容がトレースされることはもちろんです。

この性質を逆に利用して、部分トレースの場合のトレース終了の操作を行なっています。部分トレースを



注3. 正しい動作をトレースできなくとも、ステップを遡る目的ならば充分意味があります。

注4. ただし、いったん登録したら、トレーサを再ロードしない限り、解除はできません。

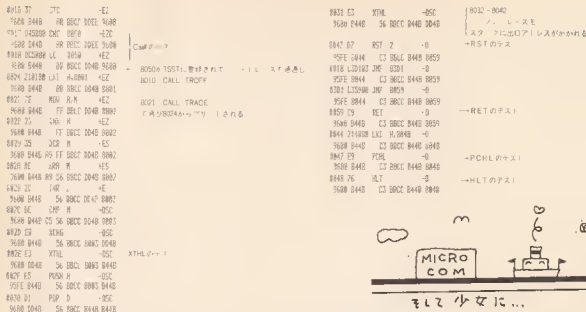
注5. NESTされている下位のサブルーチンが、標準のもとり方をしないものであっても最終的に最上位のCallの次に戻ってくれば、全体をNon Trace Modeとして、なんらさしつかえありません。

きなプログラムを作ったのです。また、テレビ・タイプライタ(クリアキー、カーソル・コントロールキー、カラー指定キー、リバーシ指定キー、グラフィック指定キー)・チェス・クロック、カセット・テープによる高速リローダー・プログラム・ロードなどのゲーム・プログラムや実用的なプログラムがありますので、他の方々にも試用してもらいたいです。

(原田さんと連絡をとりたい方は、編集部に御連絡ください。一編一)

(電局区 原田吉吉)

図2 8000からRunして8016からトレースモードになった場合



行なうには、ユーザー・プログラムを若干変更する必要があり、トレースを開始しないアドレスの直前にCALL TRACE (CD069A)をおきます。このとき全レジスタ類は、そのままトレースに引継がれます。またトレースを終了させたいステップの直後にCALL TROFF (CD109A)をおきます。このときHLを除いた全レジスタ類はトレースからユーザー・プログラムに引継がれます。HLが必要なときは、PUSH H, CALL TROFF, POP Hとしてください。<sup>注6</sup> この2つのCALLにはさまれた部分がプリント・アウトされます。1つのユーザー・プログラムで何処所にトレース・モードのON, OFFを行なっても自由ですが、フローが分岐しているときは、PUSH, POPと同様の考え方で、Pairに留意してください。<sup>注7</sup>

メモリ・ダンパ・プログラムや逆アセンブラは、自分自身をダンパさせたり、逆アセンブルさせても一応まともに完結しますが、トレースが自分をトレースしだすとLOOPに入ってしまう、正しくトレースされなくなります。これはトレースがユーザー・プログラムとは別個のスタック・ポインタとかレジスタ・セーブ・エリアを持つもののがゴツツちゃんになってしまうためです。さらに、トレースと競合するI/Oについても一般にうまくいきません。例えばユーザー・プログラムでLED DISPLAYを使っている場合、トレースの方が優先してしまい正しい表示はできません。プリントについては、プリンタのポート・インシャライズ

を含むサブルーチンをTSSTに登録してノー・トレースで実行させた場合のみ、正しく働きます。

## 使い方

以下にオペレーション・マニュアルを解説書スタイルで掲げます。ストップとしてあるのは、いずれもキーイン待ちのディスプレイ・ストップです。

- ①トレース開始アドレスを指定して、トレースを行なう場合、レジスタ類に初期値を与える必要があれば、モニターのレジスタ・セーブ・エリア (§83E 4など)に入れておき§9 A00からRUNすれば②に行く。
- ②ユーザー・プログラムの途中からトレースするときには前述の方法でCALL TRACE命令を挿入し、ユーザー・プログラムのスタート手順通りにRUNすれば挿入箇所まで実行して③に行く。
- ③LEDの左4桁(LDと略す)にLBの値 (§1000)、LEDの右4桁(RDと略す)にUBの値 (§E000)を示して止まる。LBキーを押したければ、数値をキーインし正しく入ったら[WRITE]キーを押すとLBの値が登録され、次にUBの値を変更したければ同様に操作する。いずれも[WRITE]キーを押した時点でLEDに表示されている値が登録される。<sup>注8</sup> 2回の[WRITE]キー操作で④に移る。
- ④LDにたとえば8000, RDに0000を示して止まる。LDにトレース開始アドレスを与えて[WRITE]キーを

注6. TROFFエントリ (§9 A10), TSSTに登録済みなのでユーザーが登録する必要はありません。

注7. もし誤ってトレース・モードの状態からCALL TRACEを行なっても、4ステップで脱出するようにしてありますから被害は軽微です。ただし、HLは保証されません。

注8. LB≧UBとなる値を与えても別にエラーとはならず、すべてのサブルーチンがノー・トレース・モードになる。



図4 EI, DIのテスト

```

E000 FB EI +EZ
83C7 82C8 05 D565 9565 5565
E001 2181E1 LXI H,E181 +EZ
83C7 82C8 05 D565 9565 E181
E004 227804 SHLD B478 +EZ
83C7 82C8 05 D565 9565 E181
E007 3E16 MOV R,16 +EZ
83C7 82C8 16 D565 9565 E181
E009 227804 STA B478 +EZ
83C7 82C8 16 D565 9565 E181
E00C F3 DI +EZ
83C7 82C8 16 D565 9565 E181
E00D C052FA CALL F852 +EZ
83C7 82C8 16 D565 9565 E181
E018 FB EI +EZ
83C7 82C8 16 D565 9565 E181
E011 C046F9 CALL F946 +EZ
83C5 E814 05 D565 9565 5565

```

図5 IN, OUTのテスト

```

90DE 3E99 MOV R,99 +EZ
85FE 9EF0 99 2865 9565 9088
90EB D3F8 OUT FB +EZ
85FE 9EF0 99 2865 9565 9088
98E2 7E MOV R,R +EZ
85FE 9EF0 39 2865 9565 9088
90E3 D3F9 OUT F9 +EZ
85FE 9EF0 39 2865 9565 9088
90E5 D8FA IN FR +EZ
85FE 9EF0 DF 2865 9565 9088
90E7 8F RRC +EZC
85FE 9EF0 DF 2865 9565 9088
90E8 D8E59D JC 90E5 +EZC
85FE 9EF0 DF 2865 9565 9088
90E5 D8FA IN FR +EZC
85FE 9EF0 DF 2865 9565 9088

```



に、59DE7~9FC1がトレース・アナライザにあてられている。残り6210バイトには59FFFからトレース・スタック・エリアがとられるが、トレースのレジスタ・セーブ・エリアとして1410バイト、その下にトレース・ルーチンのスタックがとられる。これは最も深いPush, Downで8レベル16バイトである。ゆえにTSSTには62-30=32バイト分すなわち16エントリ設定できる。TSSTは59FC2以上に追加される。

④このトレース・プログラムのプリント・エリアには、イニシャル・セット・ルーチンを含んでいるので、トレースをカセット・テープなどに保存しようとする方は、必ずRUNする以前のプログラムを記録することが必要で、RUNしたあとのものを記録すると解説

書どおりの機能は保証されない。

## あわりに

以上でオペレーションの説明をおわり、次にいくつかのトレースの実例を掲げておきます。(図2~図5)でトレースでリカーシブ・プログラム(CALL命令のなから自分自身を再びCallする使い方)をトレースさせると、その動きがよくわかって面白いと思います。

いろいろな命令の使われ方のすべてをトレースして試験することは、とても力がおよばず、虫が潜んでいることと思います。改訂を心掛けておりますから不都合なところを見えられた方は編集部を通じておしらせください。

TK-80トレース・プログラム V3.1 (ORG 9A00)

|             |             |             |            |                                            |            |                                   |                 |
|-------------|-------------|-------------|------------|--------------------------------------------|------------|-----------------------------------|-----------------|
| 9080 22FE9F | SHLD 9FFC   | 902C 22EC83 | SHLD 83E1  | 9063 F2489A                                | JP 9040    | 908C C05890                       | JZ 9080         |
| 9083 C0B790 | JMP 9047    | 902F E1     | POP H      | *AFTER EXECUTE ONE INSTRUCTION RETURN HERE |            | 908D 27                           | JNZ H           |
| 9086 22FE9F | SHLD 9FFC   | 9030 22F090 | CALL 90F8  | 9066 21E690                                | LXI H,90E6 | 908E 0648                         | SUI 48          |
| 9089 E1     | POP H       | 9033 E7     | PUSH R     | 9069 012848                                | LXI B,4828 | 9090 D28090                       | JNC 9080        |
| 9092 223998 | SHLD 98C9   | 9036 C0409C | CALL 9C80  | 906C 71                                    | MOV R,C    | 9093 80                           | DCR C           |
| 9095 C03890 | JMP 9087    | 9039 E5     | PUSH H     | 906D 2B                                    | DCR H      | 9094 C28190                       | J1C 9083        |
| 9098 E1     | POP H       | 903C 218888 | LXI H,0808 | 906E 85                                    | DCR B      | 9097 BF                           | CMP R           |
| 909B 218399 | LJLD 98C9   | 903F 22EE9C | SHLD 83EE  | 906F C26C90                                | JNZ 906C   | 9098 C03890                       | JMP 9088        |
| 90A4 E9     | PCAL        | 9042 218888 | LXI H,0808 | 9072 23                                    | EA H       | *FIND WATCHED CODE                |                 |
| 90A7 C04898 | LXI H,0808  | 9045 22EE9C | SHLD 83EE  | 9075 22E890                                | SHLD 98E8  | *B: OPCODE                        | A: CONTROL BYTE |
| 90AA C0     | DMP SP      | 9048 C0F048 | CALL 98F8  | 9078 283990                                | LJLD 98C9  | *RL: TOP OF MNEMONIC DATA ADDRESS |                 |
| 90AD 22FE9F | SHLD 9FFC   | 904B 223998 | SHLD 98C9  | 9079 C0E190                                | CALL 98E1  | *CODE: INVALID                    |                 |
| 90B0 C0F19F | LXI SP,9FFC | 904E 218888 | LXI H,0808 | 907C 46                                    | MOV R,B    | 9080 223890                       | SHLD 98E6       |
| 90B3 E5     | PUSH B      | 9050 22EC83 | SHLD 83EE  | 907D E5                                    | PUSH H     | 9081 E1                           | POP H           |
| 90B6 E5     | PUSH B      | 9053 C0409C | CALL 9C80  | 907E 21459C                                | LXI H,9C10 | 9082 E5                           | PUSH H          |
| 90B9 F5     | PUSH PSW    | 9056 22C09F | SHLD 9FC2  | 9081 8E39                                  | MOV C,39   | 9083 57                           | MOV R,R         |
| 90BC 28     | DCR SP      | 9059 7C     | MOV R,H    | 9084 2B                                    | MOV R,B    | 9084 87                           | RLC             |
| 90BF 28     | DCR SP      | 905B 85     | MOV L      | 9085 23                                    | JNZ H      | 9085 87                           | RLC             |
| 90C2 E1     | POP H       | 905E C06090 | JZ 9066    | 9086 BE                                    | CMP R      | *COMMAND EDIT                     |                 |
| 90C5 22EE9C | SHLD 83EE   | 9060 215498 | LXI H,9054 | 9087 23                                    | JNZ H      | 9083 F5                           | PUSH PSW        |
| 90C8 E5     | PUSH H      | 9063 34     | JNR H      | 9088 7E                                    | MOV R,R    | 9084 7E                           | MOV R,H         |
| 90CB 30     | DCR SP      | 9066 F4     | JNR H      | 9089 23                                    | JNZ H      | 9085 C0F390                       | CALL 98F3       |
| 90CE 36     | DCR SP      | 9069 3EE8   | MOV R,EE8  |                                            |            |                                   |                 |
| 90D1 E1     | POP H       | 906C 8E     | CMP H      |                                            |            |                                   |                 |

|                                                                           |            |                                 |            |                            |               |                                 |            |
|---------------------------------------------------------------------------|------------|---------------------------------|------------|----------------------------|---------------|---------------------------------|------------|
| 9808 F1                                                                   | POP PSW    | 980C F0198                      | JM 9811    | 9808 0F                    | RRC           | 9C18 3E92                       | MJ1 A,92   |
| 9809 0640                                                                 | SUI 48     | 980F C687                       | ROI 87     | 980C 0F                    | RRC           | 9C1A 03FB                       | OUT F8     |
| 980B 23                                                                   | JMK H      | 9811 4F                         | MOV C,A    | 9800 0F                    | RRC           | 9C1C C0A01                      | CALL 01A1  |
| 980E 02839A                                                               | JMC 9803   | 9812 05                         | PUSH H     | 980E 4F                    | MOV C,A       | 9C1F C01682                     | CALL 0216  |
| 980F 3E83                                                                 | MOV A,B3   | 9813 21E89A                     | LJLD 98ED  | 980F 01                    | POP H         | 9C22 47                         | MOV B,A    |
| 9811 32E89A                                                               | STB 98ED   | 9816 71                         | MOV A,C    | 980E 05                    | PUSH H        | 9C23 E619                       | RM1 10     |
| 9814 7A                                                                   | MOV H,0    | 9817 C8E89A                     | JMP 98ED   | 9811 23                    | JMK H         | 9C25 C3                         | RET        |
| <b>* MEMORICAD CVT</b>                                                    |            |                                 |            |                            |               |                                 |            |
| 9805 210008                                                               | LXI H,8888 | <b>* JUMP RETURN &amp; CALL</b> |            | 9812 3D                    | DCR A         | 9C26 29                         | DND H      |
| 9808 48                                                                   | MOV C,H    | 9818 213D9C                     | LXI H,9C3D | 9810 87                    | RLC           | 9C27 29                         | DND H      |
| 9809 C0129B                                                               | CALL 9812  | 981D 87                         | RLC        | 981E C0539B                | CALL 9853     | 9C28 29                         | DND H      |
| 980C 0648                                                                 | SUI 48     | 9821 23                         | JMK H      | 981F 23                    | JMK H         | 9C29 29                         | DND H      |
| 980E 23                                                                   | JMK H      | 9822 C8559B                     | CALL 9855  | 9810 8B                    | MOV A,H       | 9C2A 7D                         | MOV A,L    |
| 980F 02839A                                                               | JMC 9808   | 9825 C0D19B                     | CALL 98D1  | 981C 7E                    | MOV A,H       | 9C2B 68                         | DND H      |
| 98C2 7A                                                                   | MOV A,D    | 9825 C2779B                     | JMP 9887   | 981C 7E                    | CALL 98F3     | 9C2C 6F                         | MOV A,L    |
| 98C3 4A                                                                   | MOV C,D    | <b>* HSN SP &amp; DON PSW</b>   |            | 981C 7E                    | JMK H         | 9C2D C9                         | RET        |
| 98C4 E647                                                                 | RM1 87     | 9828 1F                         | RJR        | 9818 7B                    | MOV A,E       | <b>* CONSTANT TABLE</b>         |            |
| 98C5 FF81                                                                 | CPI 91     | 982C 21369C                     | LXI H,9C36 | 9811 C8F39A                | CALL 98F3     | 9C2E 42 43 44 45 46 4C 4D 4E    |            |
| 98C8 C4019B                                                               | CALL 98D1  | 982F 1E91                       | MOV E,01   | 9814 22339A                | SJLD 9839     | 9C36 42 44 4D 53 58 53 57 4E    |            |
| 98CB 21039A                                                               | LXI H,9809 | 9831 57                         | MOV D,H    | 9817 C3E79A                | JMP 98E7      | 9C3E 58 5A 5B 4E 43 43 28 58    |            |
| 98C8 85                                                                   | ADD L      | 9834 85                         | ADD L      | 9818 C0D99B                | CALL 98D9     | 9C46 4E 58 4D 58 28 4D 28 FF    |            |
| 98C8 6F                                                                   | MOV L,A    | 9837 6F                         | MOV L,A    | <b>* TEST MAKE STOP</b>    |               |                                 |            |
| 98D0 3E38                                                                 | MOV A,38   | 983A 2E83                       | MOV A,B3   | 9818 28339A                | LJLD 9839     | 9C56 4D 5E 67 86 9C 4D 56       |            |
| 98D2 8A                                                                   | RMA B      | 983B 8A                         | CMP 0      | 9810 22E883                | SJLD 83EE     | 9C5E 49 6F 05 C7 50 55 53 48    |            |
| 98D3 8F                                                                   | RRC        | 983C 8A                         | CMP 0      | 9813 C0A181                | CALL 01A1     | 9C66 FF C0 68 43 4C 4C CF       |            |
| 98D4 8F                                                                   | RRC        | 9837 C2429B                     | JMC 9842   | 9816 3E92                  | MOV A,92      | 9C6E C1 87 58 4F 5F FF C3 88    |            |
| 98D5 8F                                                                   | RRC        | 983A 1C                         | JNR E      | 9818 03FB                  | OUT FB        | 9C7E 48 4D 58 C7 64 21 43 C7    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            |                                 |            |                            |               |                                 |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 983B 3C                         | JNR A      | 9818 C02382                | CALL 0223     | 9C7E C2 21 4A FF C3 88 52 45    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 983C 81                         | RNP C      | 9818 3C                    | JNR A         | 9C86 54 C7 C8 01 52 CF 81 8A    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 983D C9429B                     | JZ 9842    | 981C 32889A                | JNZ 9838      | 9C86 4C 58 49 F8 86 41 44       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9840 1C                         | JNR E      | 981C 3A839B                | LJLD 9803     | 9C96 44 F8 86 53 55 42 F8       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9841 2C                         | JNR L      | 9814 FF48                  | CPI 48        | 9C96 88 86 41 4F 41 F8 86 86    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9842 C0559B                     | CALL 9855  | 9814 C8389A                | JZ 9838       | 9C96 58 52 41 F8 86 86 4F 52    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9845 18                         | DCR E      | 9817 FE3F                  | CPI 3F        | 9C96 41 F8 86 86 43 4D 58 C7    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9846 23                         | JMK H      | 9818 C8389A                | JZ 9838       | 9C96 84 84 49 4E 52 C7 85 84    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9847 C2429B                     | JNZ 9842   | 981C C3889A                | JMP 9866      | 9C96 44 43 52 F8 86 86 41 44    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9848 7F                         | MOV A,C    | <b>* CUSOL SE17</b>        |               |                                 |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9848 E688                       | RM1 88     | 9811 F5                    | PUSH PSW      | 9CCE 28 48 4C 48 4C 4F 38       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9849 C45C9B                     | CMP 985C   | 9812 3E8B                  | MOV A,88 1713 | 9CCE 08 4C 4F 22 58 53 54       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9850 C3839B                     | JMP 9887   | 9814 32E89A                | STB 98ED      | 9C96 48 4C 4F 38 86 53 54       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9853 85                         | ADD L      | 9817 F1                    | POP PSW       | 9CCE 41 FF C6 98 41 44 49 FF    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9854 6F                         | MOV L,A    | 9818 C9                    | RET           | 9CCE 05 98 41 4E 49 FF EE 98    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9855 C5                         | PUSH H     | 9819 0628                  | MOV B,28      | 9CCE 50 52 49 FF F6 98 4F 52    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9856 4E                         | MOV C,H    | 9818 3E92                  | MOV A,92      | 9CCE 49 FF FE 98 43 58 49 FF    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9857 C8129B                     | CALL 9812  | 9810 D3FB                  | OUT FB        | 9C96 D6 98 53 55 49 CF 82 C3    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9858 C1                         | POP B      | 980F 7E                    | MOV A,H       | 9C96 33 54 41 58 CF 8A C3 4C    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9859 C9                         | RET        | 9810 D3FB                  | OUT FB        | 9C96 41 58 FF 08 58 43 4C       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 985C 05                         | PUSH H     | 9812 D8FA                  | JM FA         | 9C96 48 47 CF 83 83 49 4E 58    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 985C 052E                       | CPI C,2C   | 9814 8F                    | RRC           | 9C96 CF 88 83 44 43 58 CF 89    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 985F C3379B                     | JMP 9857   | 9815 D8E296                | JC 98E2       | 9C96 83 44 41 4F FF 88 52       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9862 212E9C                     | LXI H,9C2E | 9818 D8FA                  | IN FA         | 9C96 43 CF FF 88 52 43 4F       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9865 C0539B                     | CALL 9853  | 9818 8F                    | RRC           | 9C96 FF 88 52 41 4C FF 1F       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9868 3E87                       | MOV A,B7   | 9818 D2E896                | JMC 98E8      | 9C96 88 52 41 52 FF 88 88 4E    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 986A 81                         | RMA C      | 981E 23                    | JMK H         | 9C96 4F 58 FF 03 98 4F 55 54    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 986B FE95                       | CPI 85     | 981F 05                    | DCX B         | 9C96 FF 2F 08 41 41 4F 2F       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 986D C2489B                     | JZ 9848    | 9818 C2DF9B                | JZ 988F       | 9C96 88 43 48 41 FF 37 88 53    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | <b>* MOV 2ND REGISTER</b>       |            | 9813 3E88                  | MOV B,80      | 9C96 54 43 47 FF 38 43 4D 43    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9878 C05E9B                     | CALL 985C  | 9815 D3FB                  | OUT FB        | 9C96 CF 98 41 43 43 4F FE       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9873 3E87                       | MOV A,B7   | 9817 C9                    | RET           | 9C96 98 53 42 47 C7 82 52       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9875 88                         | RMA B      | <b>* COMMAND KEYING IN</b> |               |                                 |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9876 212E9C                     | LXI H,9C2E | 9818 C0189C                | CALL 9C18     | 9C96 C3 C8 58 4A 48 4C FF 19    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9879 C0539B                     | CALL 9853  | 9818 28E883                | LJLD 83EE     | 9C96 D3 C8 58 4A 4C FF C9 C8    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 987C C3879B                     | JMP 9887   | 981C 0A                    | RNZ           | 9C96 58 42 48 4C FF F3 48 44    |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | <b>* MASTAT</b>                 |            | 981F C0269C                | CALL 9C26     | 9C96 49 FF FB 48 45 49 3F       |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 987F FC88                       | DM1 38     | 9812 22E883                | SJLD 83EE     | 9C96 3F                         |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9881 C5                         | PUSH B     | 9815 C3879B                | JMP 988F      | <b>* STACK TABLE INITIALIZE</b> |            |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9882 4F                         | MOV C,A    | 9818 C0189C                | CALL 9C18     | 9817 210888                     | LXI H,8888 |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9883 C0129B                     | CALL 9812  | 9818 28E883                | LJLD 83EE     | 9818 22E79A                     | SJLD 9FF2  |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9886 C1                         | POP B      | 981C 0A                    | RNZ           | 9818 210818                     | LXI H,1088 |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9887 3E38                       | MOV A,B8   | 981F C0269C                | CALL 9C26     | 9818 22E79A                     | SJLD 9FF4  |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 9889 81                         | RMA C      | 9812 22E883                | SJLD 83EE     | 9813 21C351                     | LXI H,51C3 |
| <b>* BIT P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub> OF DP SWIFT DOWN</b> |            | 988A 8F                         | RRC        | 9815 C3889C                | JMP 9C8B      | 9816 220883                     | SJLD 83DD  |



9809 215181 LK1 H.0151  
980B 220253 SHLD 030E  
980F 21125A LK1 H.9815  
9812 22049A SHLD 9824  
9813 220E9A SHLD 988E  
9815 E9 PCHL  
9817 218988 LK1 H.0000  
9818 22789E SHLD 9E7B  
9819 E1 POP H  
982E 11769E LK1 0.9E7B  
9831 7E MOV H.8  
9832 12 STRC D  
9833 80 DCR C  
9834 23 LK1 H  
9835 13 LK1 D  
9836 F2F190 JP 98F1  
9839 3E23 MOV H.23  
983B 32889A STR 9E8B  
983E 3E27 MOV H.C7  
9839 8B RMR 8  
9841 32839F STR 9F83  
9844 4F MOV C.0  
9845 3E2B MOV H.CD  
9847 8B CMP 0  
9848 C8119E JZ 9E11  
9849 79 MOV H.C  
984E FE24 CFI CA  
984F C2479E JNZ 9E47  
9811 28789E LK1 9E7B  
9814 44 MOV H.11  
9815 40 MOV C.L  
9816 C5 PUSH B  
9817 28F49E LK1 9E74  
9818 C8849E SHLD 9F84  
9819 C1 POP B  
981E 86F89E JC 9E6F  
9821 E5 PUSH B  
9822 28F29F LK1 9F72  
9823 C8849E SHLD 9F84  
9828 C1 POP B  
9829 22F59E JMC 9E6F

## \*TRACE SKIP TABLE SEARCH

982C 21C89A LK1 H.9E2B  
982F 38549B LK1 9E2B  
9832 8B CMP L  
9833 C88E9F JZ 9F0E  
9836 5E MOV E.H  
9837 23 JNZ H  
9838 7E LK1 H.A.M  
9839 23 LK1 H  
983A 7B MOV H.B  
983B 8B CMP B  
983C C2F29E JZ 9E2F  
983F 7B MOV H.E  
9848 8B CMP C  
9841 C56F9F JZ 9E6F  
9844 C32F9E JNP 9E2F  
9847 FE22 CFI C2  
9849 C8319F JZ 9F31  
984C FE2B CFI C8  
984E C84E9F JZ 9F4C

9E51 FE07 CP1 C7  
9E52 C88E9E JZ 9F3B  
9E53 7E MOV H.8  
9E57 FE03 CP1 C3  
9E59 C8319F JZ 9F31  
9E5E FE05 CP1 C9  
9E5E C8AC9F JZ 9F4C  
9E61 FE09 CP1 E9  
9E63 C8749F JZ 9F74  
9E66 FE7B CP1 7E  
9E68 C26F9E JNZ 9E6F  
9E6B AF JNZ H  
9E6C 32789E STR 9E7B

## \*EXECUTE VERTICAL CONTROL REGISTER

9E6F 31F69F LK1 SP.9F6F  
9E72 F1 POP PSW  
9E73 C1 POP B  
9E74 01 POP D  
9E75 E1 POP H  
9E76 F9 SPHL  
9E77 28F59F LK1 9F7E  
9E78 8B NOP  
9E7B 8B NOP  
9E7C 8B NOP  
9E7D 22F59F SHLD 9F7E  
9E7E C3879E JNP 9E87  
9E83 28F59F LK1 9F7E  
9E86 F9 SPHL  
9E87 E3 XTML  
9E88 22F59E SHLD 9F73  
9E8B E3 XTML  
9E8C 3B DCR SP  
9E8D 3B DCR SP  
9E8E E1 POP H  
9E8F 22F59E SHLD 9E9C  
9E92 F5 PUSH PSW  
9E93 21C89A LK1 H.988C  
9E96 23 JNZ SP  
9E97 22F59F SHLD 9F7C  
9E98 F1 POP PSW  
9E9B 21889B LK1 H.888B  
9E9E E5 PUSH H  
9E9F 31F59F LK1 SP.9F6F  
9E9D 8B PUSH D  
9E9E 3B PUSH B  
9E9F F5 PUSH PSW  
9E9E E1 POP H  
9E9E 31F29F LK1 SP.9F72  
9E9E E5 PUSH H  
9E9E 3E21 MOV H.C1  
9E9E C3E89A STR 9E8B  
9E9E 8E2B MOV H.C2  
9E9E 3E28 MOV H.8  
9E9E 85 RMR L  
9E9E C88E9E JZ 9E8B  
9E9E 8E2B MOV H.C2  
9E9E C8129B SHLD 9E12  
9E9E 8E45 MOV H.C45  
9E9E 3E8A MOV H.8A  
9E9E 85 RMR L  
9E9E C2C89E JNZ 9E6C

9E9A 8E4F MOV C.4F  
9E9B C8129B SHLD 9E12  
9E9C 81889A LK1 H.888B  
9E9C C88E9F SHLD 9F8B  
9E9F 81539B LK1 H.2853  
9E9E C88E9F SHLD 9F8B  
9E9E 81539A LK1 H.1859  
9E9B C88E9F SHLD 9F8B  
9E9B 814381 LK1 H.8143  
9E9E C88E9F SHLD 9F8B  
9E9E 21879B LK1 H.987B  
9E9A C88E9F SHLD 9F8B  
9E9F 3E2D MOV H.C8  
9E9F 28F59B STR 9E8B  
9E9C 28F59F SHLD 9F7C  
9E9F C8E19A SHLD 98E1  
9E9E 21889B LK1 H.888B  
9E9E C8E19A SHLD 98E1  
9E9B 38789E LK1 9E7B  
9E9B 8E6E RMR FE  
9E9B FE34 CFI 34  
9E9F C88E9F JZ 9F0F  
9E9E 3E8B MOV H.8B  
9E9A FE2B CFI 86  
9E9E C2189F JNZ 9F1E

## \*EXPRESSING DATA OF M

9F09 28F59F SHLD 9F7E  
9F0C 7E MOV H.11  
9F0D C8F39A SHLD 98F3  
9F10 3E2D MOV H.8D  
9F12 3E2B9A STR 98E8  
9F15 F1 POP PSW  
9F16 C8E19A SHLD 9F7E  
9F19 28F59F SHLD 9F7B  
9F1C C8E19A SHLD 98E1  
9F1F 28F59F SHLD 9F7B  
9F22 C8E19A SHLD 98E1  
9F25 28F59F SHLD 9F7E  
9F28 C8E19A SHLD 98E1  
9F2B 21C79B LK1 H.90C7  
9F2E C3879E JNP 988B

## \*IMP

9F31 28789F SHLD 9E7B  
9F34 22449F SHLD 9E44  
9F37 21489F LK1 H.9E48  
9F38 22769E SHLD 9E7B  
9F3D C36F9E JNP 9E6F

## \*CONDITIONAL JUMP RETURN

9F48 22F59F SHLD 9F7E  
9F43 21889B LK1 H.888B  
9F46 22399A SHLD 9839  
9F49 C3879E JNP 9E87

## \*REL

9F4C 3E21 MOV H.21  
9F4E 328E9F STR 9E8B  
9F51 28F59F SHLD 9F7C  
9F54 F9 SPHL  
9F55 E3 XTML  
9F56 22449F SHLD 9E44  
9F59 E3 XTML  
9F5B 31F69F LK1 SP.9F6F  
9F5D F1 POP PSW

9F5E C1 POP B  
9F5F D1 POP D  
9F60 E1 POP H  
9F61 E1 POP H  
9F62 318E9F LK1 SP.9E7E  
9F65 C7789E JNP 9E7B

## \*CONDITION RET RETURN

9F68 22F59F SHLD 9F7E  
9F6B 28F59F SHLD 9F7C  
9F6E F3 SPHL  
9F6F 33 LK1 SP  
9F70 33 LK1 SP  
9F71 C3439F JNP 9F43

## \*PCHL

9F74 28F59F SHLD 9F7E  
9F77 22399A SHLD 9839  
9F79 28F59F SHLD 9F7E  
9F7D C3879E JNP 9E87

## \*RST

9F80 3E38 MOV H.38  
9F82 8B RMR B  
9F83 6F MOV H.B  
9F84 97 SUB B  
9F85 67 MOV H.8  
9F86 3E2D MOV H.CD  
9F88 32789E STR 9E7B  
9F8B C3919F JNP 9E91

## \*CALL

9F8E 28789E SHLD 9E7B  
9F91 22449F SHLD 9E44  
9F94 21889F LK1 H.9F8B  
9F97 C3289E JNP 9F3B

## \*CALL TRACE END

9F98 22F59F SHLD 9F7E  
9F99 28399A SHLD 9839  
9F9B E3 XTML  
9F9C C3439F JNP 9F43  
9F9D 7B MOV H.B  
9F9E 8C JNP H  
9F9E C28E9F JZ 9F9E  
9F9B 45 MOV H.C  
9F9B 65 MOV H.L  
9F9B 7B MOV H.B  
9F9C 8C CMP H  
9F9D 8B RZ  
9F9E 8C JNZ H  
9F9F 7B MOV H.B  
9F9B F8859F JNP 9F85  
9F9B 8C JNP H  
9F9B 67 RET  
9F9B 67 RET  
9F9B 3F CMC  
9F9B 67 JNC  
9F9B 7B MOV H.8  
9F9B 85 RMR L  
9F9B C4129B SHLD 9B12  
9F9B 67 RET

※アセンブリのときは  
9890 15--00  
98A7 C8E190--21A790  
98C5 48--00  
に設定する

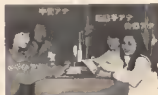
## 山梨県のみなさん！

我が山梨マイコンクラブは、本年3月末に活動を開始しました。現在会員数は女性を含め25名ですが、ビギナーからマニアの方まで広くソフト、ハードの情報交換の場として活動することを目標としています。同好の方、参加してみませんか。

N A S A コンピュータ事業部内 山梨マイコンクラブ

甲府市丸の内1-9-19 県信ビル ☎ (0552)37-7373

NASA通信機が、山梨放送で  
行なっているディスク・ジョッキーの模様。



キミのプログラム・ライブラリにぜひとも欲しい!

# LKIT-16 用 逆アセンブラ



稲岡泰宏

最近になってようやく、パナファコム社からLKIT-16の周辺装置やソフトウェアが供給されるようになりましたが、しびれを切らせてとうの昔に自作された方も多いと思います。

ところでデバッグのとき案外やっかいなのが、アセンブル・ミスの発見です。これが簡単に行なえるようにと考えたのが、この逆アセンブラです。

## 1 フォーマット

LKIT-16は元来、PFL16Aシステムのソフト・トレーニング用のようです。したがって、アセンブラのコーディングもPFL16Aに都合のいいようになっています。ところが実際のキーインにおいては、コーディングされたプログラムを頭の中で、少々変更しながら入力しなければなりません。

例. コーディング・フォーム L R0, X ▽5A ▽

実 際 の キー イ ン LR0hh5A

PFL16Aのソフトウェア・トレーニングという本来の目的で使っているならいざしらず、LKIT-16そのものシステム全景



ので楽しもうというアマチュアにとって、これは2重下問です。それならいっそのこと最初から、

L R0 hh 5 0

とコーディングしたらどうでしょうか。そういうわけで、この逆アセンブラではキーインどおり、出力することにしました。ところがASCIIには小文字がないので、

L R0 HH 5 0

となってしまいます。しかし、これだけでは不便なのでアドレスと16進の機械語をつけます。

0040\_0C050\_L\_R0\_HH\_50

アドレス 機械語

すべてキーインどおりではありますが、例外があります。それはCLEARです。

CLEARR1

などとキーインしますが、

C L E A R \_ R 1

とAが省略されて出力されます。あってもなくても同じようなものですから、かんべんしてください。

LKIT-16のアセンブラにはないけれど、逆アセンブルされる命令があります。無効命令です。(NOPではありません) 命令語の上位5bitが“0”である命令は、無効命令でCPUは停止します。(PFL16A LSI 説明書 p. 28参照) この場合は“HALT”と出力します。LKIT-16の“H”命令とは別なものです。無効命令はCPU MN1610にとって、本来のHALT命令に相当するからです。ちなみにLKIT-16の“H”命令はモニタに分歧します。プログラム実行中に全セグメントが消灯し、RESET以外きかなくなるのは、この無効命令でCPUが停止した場合がいがい多いようです。



## ② 使用法

このプログラム自体、サブルーチンになっています。CRTやPRINTERへの出力ルーチンと組み合わせて使うようになっているのです。手順を次に示します。

① OPB (OUT PUT BUFFER)。結果をしまっておくバッファとして、32語確保します。このバッファの下位バイトがASCIIコードで出力されます。上位バイトに何が入るかは、当方一切関知しません。このOPBの先頭番地を“X1”に入れます。

② 対象となる語のアドレスを“X0”に入れます。このプログラムは、1回のコールで1語ずつ逆アセンブルしますので、続けて行なう場合には、“X0”に1加えて再びコールします。ただし、OPBには前回のコールで得た結果は保存されず、新しい結果が入るので注意してください。

この2つのパラメータを与えれば動作しますが、スタックの問題があります。

作業領域はスタック上に展開していますから、スタックは17レベル使用します。これもちゃんと確保してください。

このプログラムは相対番地、あるいはそれに準じる方法でプログラムしてあります。ですからメモリのどこにでも移せます。ROM化も可能なはずです。

### ※デバッグ

SKIP命令を逆アセンブルすると、レジスタとスキップ条件の間にスペースが入りません。これを訂正するには、91番地と92番地の間に、

```
AI X1 1
```

を挿入すればよいのですが、これをやるには、以下のX0によりインデックス修飾された命令を、ほとんど訂正しなければなりません。どうしても気になる方は挑戦してください。

### 出力フォーマット例

| アドレス | 候補語                   |
|------|-----------------------|
| 1100 | CF1A__B__*_+_1A       |
| 11BC | 6101__CLER_R1         |
| 005F | 0900__MVI__R1__00     |
| 024A | 6201__EOR__R2__R1     |
| 101B | A734__BAL__HH(X0)__34 |
| 1001 | 2841__TBIT__R0__1_Z   |

### プログラムを書いた人



我が愛機



### ※お知らせ

全国のマイコンクラブの皆さん情報交換をしましょう。当サークルはLKIT-16をはじめ、Z80、8085、4004、6800、SC/MPなど、もろもろのプロセッサを扱っています。

〒602 京都市上京区烏丸通今出川東入る

同志社大学工学部内H-513

電気研究会      マイコンコンピュータ班

### +++++つねれもの

DC命令は、逆アセンブルできません。プログラムと定数の区別がないからです。むりにすればプログラムとして扱われます。DC命令の逆アセンブルはメモリ・ダンプにまかせます。

### □参考文献

LKIT-16ユーザーズマニュアル

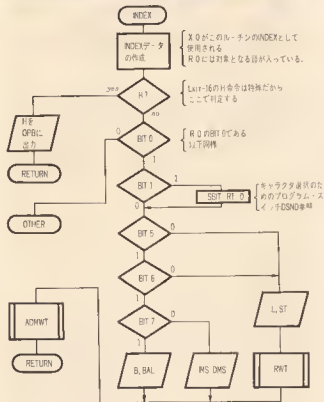
LKIT-16プログラミングマニュアル

PFL16A LS1説明書

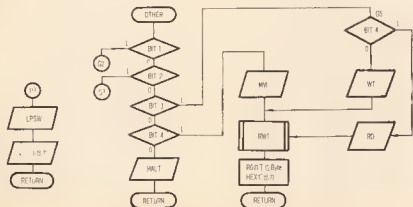


### I/Oプラザ

▲先日、他誌の西武デパートにAPPLE IIが置いてあるという話をきいて見に行きました。カラーTVを使って模倣みたいなのを売っていました。ところがようどPETの展開を待たせて、ここで、はじめてBAS ICとスタートレックをやらせてもらいました。BIT 1NNなんかでBAS ICが使えそうになっているんだけどどうでも人だていっばいなんだよね。あれどうなってんのかな? スタートレックもどこか無料で作らせてく

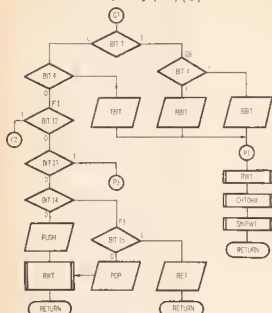


フローチャート(3)

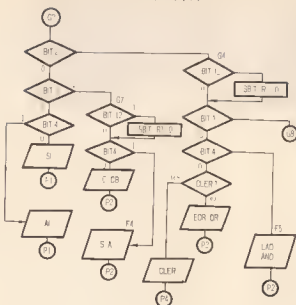


れるところないの? あったら教えて、教えて、(と)ここでスタートレックに負けました、くやしい、ぜひもう一度やり  
たい、) ☆どうもスタートレックっていうのは単調ですね、なんかもっとおもしろいものがないかな、(APPLE  
のスタートレックってどんなんですか? 誰かやらせて、) とにかくあの単調さから抜け出さないでなくてはなら  
ない、) 国に習のメモリを使う場合、機械語とBAS1Cどちらが複雑になるのか? 単純に機械語ってきえていいの

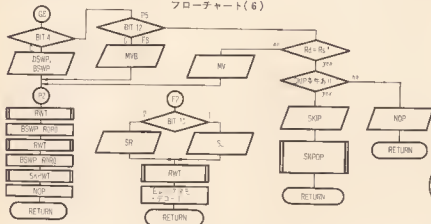
フローチャート(4)



フローチャート(5)



フローチャート(6)



キャラクタ・テーブル・リスト

11C3 4D54 5642 4949 2D54  
 11C7 5752 5444 5D5D 4F55  
 11C9 5657 3846 5353 524C  
 11CF 324C 4558 5453 2B57  
 11D3 5253 4242 4943 5454  
 11D7 5341 4949 4343 423D  
 11D9 5341 2B2D 434F 4F52  
 11DF 522D 4141 414E 4444  
 11E3 4442 5353 5757 5D5D  
 11E7 4D4D 5456 422D 534C  
 11E6 542E 4242 412D 4C2E  
 11EF 4449 4D4D 5353 484E  
 11F3 2B4F 2B5E 4552 4C4E

11F7 4E4F 515D 5C5C 3A51  
 11FB 525F 7D7D 5D53 715D  
 11FF 572D 542B 522B 2B5C  
 1203 2B4E 2B5D 425E 2B5D  
 1207 5B4E 0C5F 4B5C 562F  
 1209 5445 5B4E 2B5D 4C4F  
 120F 544E 2B5D 4F4F 4B5E  
 1213 5F2D 4F4E 5B4D 5B2D  
 1217 2B5E 2B4F 534D 4545  
 121B 4D7E 4B2D 2B2D 4B2D  
 121F 4C2D 4B2D 4B4C 4B4C  
 1223 2B2D 5B5E 3B51 2A2D  
 1227 2B2D 4B4F 4B4D 2B2D

122B 2E2D 5D5D 7B51 2B2D  
 122F 2B2D 4B44 4B47 4B2D  
 1233 542E

タコ焼

かな、16Kビッシリ機械流でスタートレック作っただけならもっとおもしろくなるのかしらん、それにしてもPETはハンサムですね、TRS-80や、APPLE IIもいいけど、もしかしらた僕のはLKT-16よりも……やっぱりハンサムな方がいいな、だからエンタープライズよりも宇宙戦艦ヤマトの方がいいのだ(みんなで宇宙戦艦ヤマトゲームを作ろうではないか、僕は何をいいたかったのかな?) (松戸市 中谷由紀夫)

## 逆アセンブラ・リスト

BEGIN

ADD

MACHIN

RETURN

AMWT

BWT

SKWWT

DO PUSH1  
AMWT

AMWT

INDEX

CHTOKX

テン ト 3 出力

CHTOKA

テ ト 2 出力

CHTOKX

CHTOKX

DSNOZ

DSNOI

DSNOZ

DSNOZ

CHTOKX

IF A > B  
GOTO NEXT

SUB4

CHC37 桁並テーク

DSNOI

CHTOKX

CHTOKX

R2 の下位バイトの  
2 の乗数をとる

SD05

CHTOKX

CHTOKA

INDEX

IC → AO

SAPDP

BWT

+ BWT

SAPWT

+ SKPWT



05. \* CTR0X サブル-チャンネル  
06. \* SUB0 中継  
07. \* SUNS  
08. 1000 1000  
09. 1000 1000  
10. 1000 1000  
11. 1000 1000  
12. 1000 1000  
13. 1000 1000  
14. 1000 1000  
15. 1000 1000  
16. 1000 1000  
17. 1000 1000  
18. 1000 1000  
19. 1000 1000  
20. 1000 1000  
21. 1000 1000  
22. 1000 1000  
23. 1000 1000  
24. 1000 1000  
25. 1000 1000  
26. 1000 1000  
27. 1000 1000  
28. 1000 1000  
29. 1000 1000  
30. 1000 1000  
31. 1000 1000  
32. 1000 1000  
33. 1000 1000  
34. 1000 1000  
35. 1000 1000  
36. 1000 1000  
37. 1000 1000  
38. 1000 1000  
39. 1000 1000  
40. 1000 1000  
41. 1000 1000  
42. 1000 1000  
43. 1000 1000  
44. 1000 1000  
45. 1000 1000  
46. 1000 1000  
47. 1000 1000  
48. 1000 1000  
49. 1000 1000  
50. 1000 1000  
51. 1000 1000  
52. 1000 1000  
53. 1000 1000  
54. 1000 1000  
55. 1000 1000  
56. 1000 1000  
57. 1000 1000  
58. 1000 1000  
59. 1000 1000  
60. 1000 1000  
61. 1000 1000  
62. 1000 1000  
63. 1000 1000  
64. 1000 1000  
65. 1000 1000  
66. 1000 1000  
67. 1000 1000  
68. 1000 1000  
69. 1000 1000  
70. 1000 1000  
71. 1000 1000  
72. 1000 1000  
73. 1000 1000  
74. 1000 1000  
75. 1000 1000  
76. 1000 1000  
77. 1000 1000  
78. 1000 1000  
79. 1000 1000  
80. 1000 1000  
81. 1000 1000  
82. 1000 1000  
83. 1000 1000  
84. 1000 1000  
85. 1000 1000  
86. 1000 1000  
87. 1000 1000  
88. 1000 1000  
89. 1000 1000  
90. 1000 1000  
91. 1000 1000  
92. 1000 1000  
93. 1000 1000  
94. 1000 1000  
95. 1000 1000  
96. 1000 1000  
97. 1000 1000  
98. 1000 1000  
99. 1000 1000  
100. 1000 1000



01. 1000 1000  
02. 1000 1000  
03. 1000 1000  
04. 1000 1000  
05. 1000 1000  
06. 1000 1000  
07. 1000 1000  
08. 1000 1000  
09. 1000 1000  
10. 1000 1000  
11. 1000 1000  
12. 1000 1000  
13. 1000 1000  
14. 1000 1000  
15. 1000 1000  
16. 1000 1000  
17. 1000 1000  
18. 1000 1000  
19. 1000 1000  
20. 1000 1000  
21. 1000 1000  
22. 1000 1000  
23. 1000 1000  
24. 1000 1000  
25. 1000 1000  
26. 1000 1000  
27. 1000 1000  
28. 1000 1000  
29. 1000 1000  
30. 1000 1000  
31. 1000 1000  
32. 1000 1000  
33. 1000 1000  
34. 1000 1000  
35. 1000 1000  
36. 1000 1000  
37. 1000 1000  
38. 1000 1000  
39. 1000 1000  
40. 1000 1000  
41. 1000 1000  
42. 1000 1000  
43. 1000 1000  
44. 1000 1000  
45. 1000 1000  
46. 1000 1000  
47. 1000 1000  
48. 1000 1000  
49. 1000 1000  
50. 1000 1000  
51. 1000 1000  
52. 1000 1000  
53. 1000 1000  
54. 1000 1000  
55. 1000 1000  
56. 1000 1000  
57. 1000 1000  
58. 1000 1000  
59. 1000 1000  
60. 1000 1000  
61. 1000 1000  
62. 1000 1000  
63. 1000 1000  
64. 1000 1000  
65. 1000 1000  
66. 1000 1000  
67. 1000 1000  
68. 1000 1000  
69. 1000 1000  
70. 1000 1000  
71. 1000 1000  
72. 1000 1000  
73. 1000 1000  
74. 1000 1000  
75. 1000 1000  
76. 1000 1000  
77. 1000 1000  
78. 1000 1000  
79. 1000 1000  
80. 1000 1000  
81. 1000 1000  
82. 1000 1000  
83. 1000 1000  
84. 1000 1000  
85. 1000 1000  
86. 1000 1000  
87. 1000 1000  
88. 1000 1000  
89. 1000 1000  
90. 1000 1000  
91. 1000 1000  
92. 1000 1000  
93. 1000 1000  
94. 1000 1000  
95. 1000 1000  
96. 1000 1000  
97. 1000 1000  
98. 1000 1000  
99. 1000 1000  
100. 1000 1000





**RWT (1034番地)**

R 0 内にある命令のレジスタを評価して、OP B に ASCII コードに変換し、出力します。

R 0 の Bit 5, 6, 7 が対象ですから R<sub>0</sub> を評価する場合には B SWP して用います。

X 1 には適当な OP B の位置が与えられていなければなりません。

例. Rd X 1 の初期値 + 16n → X 1

R<sub>0</sub> X 1 の初期値 + 20n → X 1

R 0 は保存, R 1 は破壊, その他は保存されます。

使用サブルーチン

DSND1, DSND 2

スタック

RWT 自身 1 レベル

DSND 1, 2 6 レベル

計 7 レベル

**SKPWT (1044番地)**

スキップ条件を評価します。

X 1 に適当な OP B の位置が与えられていることが必要です。

R 0, R 1, R 2, E レジスタが破壊されます。X 1 はキャラクター数だけ増加

使用 SUB ルーチン DSND 2 (116A 番地)

DSND 3 (116C 番地)

スタック... 7 レベル使用。

**SUB 4 (1072番地)**

相対番地指定において、アドレス部を、符号付き 16 進数にします。

その結果を OP B に入れます。

使用 SUB ルーチン

CHTOHX (1098 番地), DSND 1 (1168 番地)

R 0 の下位 Byte 対象

X 0 のみ保存

**SUB 5 (1086番地)**

アドレス部の 16 進内部コードを ASCII 16 進数にします。

その結果は OP B に入ります

**INDEX (108E 番地)**

逆アセンブラの最も MAIN なルーチンです。

A D D 欄の数字は X 0 によりインデックス修飾するためのアドレスです。

**ADMWT (1170番地)**

アドレッシング・モードを評価して、ASCII コードで OP B に出力します。

X 1 にはあらかじめ、OP B の適当な位置が与えられていなければなりません。

このモードキャラクターの 1 スペース後にアドレス部が出力されます。

例. \*...+ 1 F

HH... 3 A

(\*)... 0 F

R 1 は破壊され、X 1 はキャラクター数だけ増加。

その他は保存。

スタック... 7 レベル使用

**DSND(n) n = 1, 2, 3, 4 (1197 番地)**

DATA SEND (n)

n 語の DATA を転送する。

転送先 X 1 の内容の示す番地から連続する n 語

転送元 SUB 1 ルーチンの POP 命令から、R 1 の下位バイトの内容だけ離れたところにある連続する n 語。

使用法

キャラクター・テーブルの上位バイトを転送する場合は、R 1 の Bit 0 を "0", 下位の場合は "1" としておく。

R 1 の下位 Byte には「転送元」に示されている数 (POP と転送元の距離) を入れる。

X 1 は OP B の先頭番地を示すものであるが、これに T A B 数を加えておく

| BIT | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

0 のときは 1 Byte  
1 のときは 2 Byte

キャラクター・テーブルと SUB 1 の  
POP 命令との相対距離

つまり Bit 0 = 0 のときは 1 Byte  
と下位 Byte が反転されて転送される

パラメータ... R 1

スタック... 6 レベル使用

DSND 1 ~ SUB 3 までは連続した番地に入れます。

X 1 以外のレジスタ保存。

X 1 は転送語数だけ増加

**CHARACTOR TABLE (111C 番地)**

OP B 出力用のキャラクター・テーブルである。

サブルーチン "DSND(n)" によって OP B に転送される。

COMMENT 欄の数字は、DSND(n) のサブルーチン 1 の P OP 命令との相対距離である。

OP, CODE はすべて "DC"

LABEL の C H C は、"CHARACTOR CODE" の略です。

# Book Guide



JMC 編集 『マイコンコンピュータ教科書』

(JMC 3 級)

本書はマイコンコンピュータ技術者養成を行なっている、JMC スクール 3 級の標準教科書で、8080, 8085, および Z80 についての基本説明から、プログラムまで解説してある。内容は論理回路の基礎から応用まで述べられており、初心者から高度な技術者まで利用できる。

日本マイコンコンピュータ社 ￥4,500

H68/TR

## HIT &amp; BLOW

## 数あてゲーム

北原 毅

マイクロコンピュータと言っても、いろいろありますが、入出力装置のまったくないもの、もしくは極めて簡単なものは、プロのように機軸に組み込んで使用するようなことのないアマチュアにとっては、もの足りない気がします。

このような実状を考慮して、各メーカーでも、いろいろ周辺装置を発売し始めています。今回はその中で、日立から発売になったテレビ・インターフェイス・モジュールH68/TV-01を使ってみました。

## 特 徴

H68/TV-01は日立のトレーニング・モジュールに接続して、家庭用テレビやモニター・テレビをディスプレイ端末として使用するためのインターフェイスで、次のような特徴を持っています。

- ①文字表示のほか、グラフィック表示として128×96ドットが使える。
- ②文字表示は家庭用テレビで32文字×16行、モニター・テレビで64文字×16行まで表示できる。
- ③BASICが使える。

などです。では、少し詳しく中身を説明していきましょう。

## ハードウェア

メモリマップを図1に示します。斜線で囲ったところが、TVモジュールの部分です。また、ブロックダイアグラムを図2に示します。

これだけで、だいたいの特徴は理解していただけたと思いますが、2つ3つ重要なところを上げておきます。

まず、4KのRAM(\$1000~\$1FFF)は、書き込み禁止が手動(スイッチ切換え)でできるようになっています。BASICなどを入れておいて、MPUが暴走しても破壊を防ぐことができます。

白黒反転ができること、これは文字単位でなく、全画面を反転させます。グラフィックなどを使うときには便利ですよ。

ステータス情報フラグ(\$E0A8)。これが、一番の特徴ではないかと思います。ダイアグラム中にあるように、リフレッシュ・メモリは、CRTCとMPUの両方からアクセスされます。普通のV-RAMだとMPU優先で、MPUにアクセスされているときは、その内容を画面に表示しないようになっています。

このモジュールは逆で、CRTCが垂直の掃線期間、すな



わち画面表示していないとき以外はMPUからアクセスできないようになっています。その掃線期間中であるかどうかを示すのが、\$E0A8番地のD7で、たとえば、Aレジスタの内容をリフレッシュ・メモリの\$B000番地に書き込むためには、次のようなプログラムになります。

(注) \$○○○○は16進数を表わす。

|     |      |        |               |
|-----|------|--------|---------------|
| L01 | TST  | \$E0A8 | ステータス・フラグを調べる |
|     | BMI  | L01    | 掃線期間なら次へ      |
|     | STAA | \$B000 |               |
|     | TST  | \$E0A8 | ステータス・フラグを調べる |
|     | BMI  | L01    | 掃線期間なら次へ      |
|     | SWI  |        |               |

このように、2度ステータス・フラグを調べる必要があります。STAA実行中に掃線期間を抜け出すと、正しいデータの書き込みは保証されないからです。

CRTCの使用法については、マニュアルに詳しく書かれていますが、ライトペンは一応使えないようになっています。ただ、内部のレジスタへの書き込み方が、変わっているのので少し説明します。

このモジュールでは、\$E0A0番地にCRTCの内部レジスタの番号(R10なら\$0A)を書き込み、その次に\$E0A0番地の示すレジスタに、書きたい値を書き込みます。内部レジスタをイニシャライズする場合、この操作を10回から15回ほどくり返します。

電源は5V2Aですから、H68/TRと同一電源にするには、5A以上のものが必要となります。

H68/TRの拡張には、バス・ドライバが7つ必要であることをお忘れなく。

## ソフトウェア

ハードのほうはこのくらいで、ソフトウェアの内容に移りましょう。

図2 ブロック・ダイアグラム

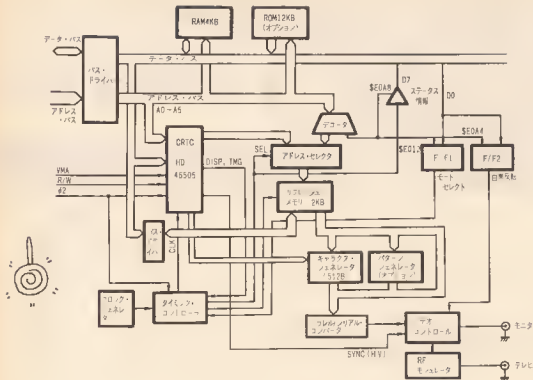
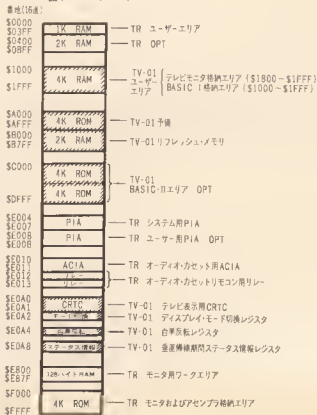


図1 メモリ・マップ



TV-01に、カセットテープで付属してくるソフトは、次の4つです。

- ①機能確認プログラム \$0000～\$00A5
- ②テレビモニタ \$1800～\$1FFF
- ③テレビサブルーチン \$1000～\$14FF
- ④BASIC-I \$1000～\$1FFF

まず、機能確認プログラムでは、64文字表示とフルグラフィック（チェッカーマークになる）で、ポジ、ネガのくり返しになります。これで、CRTCのイニシャライズによる画面の修正などの方法がわかります。

モニタに移ります。2K Byteのモニタですが、この機能には驚きます。ROMに入っているモニタの機能はフルに持っていて、そのほかに、前後ページの内容はカーソルをスクロールして、画面に表示し、逆アセンブラ（TRモジュールのマニュアルにある\$0000から始まるもの）を実行すると、タイトルが出て逆アセンブルしたものが、スクローリングして15行までは残っていて見えます。

アセンブラもアドレスを同時に示してくれるので、非常に便利です。またメモリダンプの機能も持っていて、指定した番地から指定した番地まで（\$10単位）、16進2桁で表示してくれます。ただ、中断がきかないのが残念ですが、かなり便利なものです。またBASICと共存できないのも残念な気がしますが、内容を改造してROMにするつもりでいます。

テレビ・サブルーチンは、次の11本が供給されます。

- ①キャラクタ・イニシャライズ 32文字/行
- ②キャラクタ・イニシャライズ 64文字/行

## 簡易ダズラー・プログラム

| アドレス | ラベル | 命令   | コメント   | アドレス           | ラベル  | 命令   | コメント   |        |
|------|-----|------|--------|----------------|------|------|--------|--------|
| 0100 | L01 | JSR  | \$1006 | グラフィック・イニシャライズ |      | LDAB | 1, X   |        |
|      |     | LDAA | #502   |                |      | ADDB | \$FD   |        |
|      |     | STAA | \$FE   |                |      | LDAA | \$FC   |        |
|      |     | LDX  | #5B000 | リフレッシュ・メモリ市販機  |      | ADCA | #501   |        |
|      |     | STX  | \$FC   |                |      | CMPA | #5B6   |        |
|      |     | CLRB |        |                |      | BLT  | L05    |        |
|      |     | SEI  |        |                |      | LDAA | #5B0   |        |
|      |     | LDX  | #L04   | タイマ割り込み        | 0148 | L05  | STAA   | \$FC   |
|      |     | STX  | \$E804 | スタート           |      | STAB | \$FD   |        |
|      |     | LDAA | #5     |                |      | LDAB | 1, X   |        |
|      |     | STAA | \$E007 |                |      | LDAA | #5AA   |        |
|      |     | CLI  |        |                |      | BITB | 4, X   |        |
|      | L02 | LDX  | #5B000 |                |      | BEQ  | L06    |        |
|      | L03 | TST  | \$E0A8 | ステータス情報チェック    |      | COMA |        |        |
|      |     | BMI  | L03    |                | 0155 | L06  | LDAB   | #510   |
|      |     | STAA | X      |                |      | LDX  | \$FC   |        |
|      |     | TST  | \$E0A8 |                | 0159 | L07  | TST    | \$E0A8 |
|      |     | BMI  | L03    |                |      | BMI  | L07    |        |
|      |     | INX  |        |                |      | STAA | X      |        |
|      |     | ADDB | #80F   |                |      | TST  | \$E0A8 |        |
|      |     | CPX  | #5B600 |                |      | BMI  | L07    |        |
|      |     | BNE  | L03    |                |      | INX  |        |        |
|      |     | ADDA | #508   |                |      | DECB |        |        |
|      |     | BRA  | L02    |                |      | BNE  | L07    |        |
| 0134 | L04 | LDAA | \$FE   | 横方向に点線を書き入れる。  |      |      |        |        |
|      |     | DECA |        |                | 0169 | L08  | LDAA   | #502   |
|      |     | BNE  | L09    |                | 016B | L09  | STAA   | \$FE   |
|      |     | TSX  |        |                |      | JMP  | \$F00F |        |

このプログラムは、画面で縦に動く線と横に割り込む点線の作る、変わった模様を表示します。

- グラフィック・イニシャライズ
- キャラクタ・ディスプレイ・モード切替
- グラフィック・ディスプレイ・モード切替
- 相対アドレス画面表示
- 絶対アドレス画面表示
- 1ドット消去
- 1ドット表示
- 直線表示
- 点線表示

おとと



●から●までは、ほとんどCRTCのイニシャライズ・プログラムです。●は、カーソルの示す位置にパラメータで示されたASCIIコードの文字を表示するプログラム。●は、パラメータ2で指定されたアドレスに、パラメータ1のASCIIコードの文字を表示するプログラム。

●、●は、グラフィック・モードで、パラメータで指定された点を点滅させるプログラム。●、●は、かなり便利なサブルーチンで、始点と終点をパラメータで与えてやるとその2点を直線または点線で結んでくれます。

では、このサブルーチンのグラフィック・イニシャライズのプログラムを使って、簡易ダズラーをディスプレイしてみよう。

さて、いよいよBASICの機能について紹介しましょう。コマンドの一覧表は、表1のとおりです。

表1の中で、\*印のついたものだけをとりあげて少し説明し、その後実際に組んだプログラムを追いながら、補足していきなさいと思います。

**LIST** このコマンドの特徴は、長いプログラムを表示する場合、ページごとに一時停止し、**[AB]**、**[R]**、**[ES]** キー以外を入力すると、次の1ページ分を表示します。

**[RES]** は、BASIC自体から抜け出しますので別にして、**[AB]** は、このコマンド実行中に限らず、実際プログラムの実行中でも(その名のとおり)、中断させることのできるキーです。ほかの

表1 TV-01のコマンド

| コマンド          | 説明  | 注                              |
|---------------|-----|--------------------------------|
| RUN           | R   | プログラム実行                        |
| LIST          | *   | L プログラム表示(ページごとに一時停止して)くれる     |
| NEW           | N   | プログラム消去                        |
| SIZE          | *   | S 現在プログラムが使用しているエリア、残りエリアの表示   |
| CONT          | *   | C STOP文でプログラムを中断した際の再開を指示      |
| SAVE          | *   | メモリ上のプログラムをカセットに格納             |
| LOAD          | *   | カセットテープ上のプログラムをメモリへ移動          |
| MEM           | *   | 使用するメモリ・エリアを指定する               |
| EX            | --- | ---                            |
| LET           | --- | 代入、変数に値を代入する                   |
| DIM           | *   | 変数に配列をとる。                      |
| PRINT         | PR  | 変数の値等を出力。                      |
| INPUT         | IN  | 変数にキーボードから値を設定                 |
| GOTO          | GO  | 希望する文番号を持つ文に値を設定               |
| IF            | --- | 条件判定文                          |
| FOR, TO, STEP | --- | 2つで囲まれた範囲内にある文を指定された回数だけ繰り返す。  |
| NEXT          | NXT | 数だけ繰り返す。                       |
| REPEAT        | --- | 1番新しく実行されたGOSUB文の次の文に戻る。       |
| STOP          | *   | プログラムの実行を中断する。                 |
| END           | --- | プログラムの実行を終了。                   |
| REM           | --- | コメント文                          |
| POKE          | *   | 絶対番地アドレスにデータを設定                |
| CALL          | *   | 絶対番地アドレスに制御を移す                 |
| PEEK          | *   | 絶対アドレスの内容を読み出す                 |
| ABS           | --- | 絶対値をとる。                        |
| RND           | --- | 乱数を生ずる                         |
| TAB           | --- | 指定した数だけスペースを空ける                |
| CHR\$         | *   | 指定した数値の下位1バイトをASCIIコードと見て出力する。 |

**BASICのコントロール/Cにあたるものです。**  
これはプログラムを組んでいるとき、あとどれだけのメモリが残っているか、またSTOP文で中断したときに、DIM文でどれだけメモリ上限がつかってきているかなどがわかります。  
**CONT** CONTINUEの略でしょう。STOP文で中断し、変数の値など調べたあと、いま、止めた次の文番



号から実行するのにCONTを使います。ダイレクト・モードです。

**SAVE** このコマンドで特筆すべきことは、リレーのオンラインということですが、テープレコーダのリモート端子をいちいちはずす必要がありません。

**MEM** このコマンドで、プログラムの格納先頭番地と最終番地を16進で指定してやることによって、BASICで書かれるプログラムをどの番地から格納するか変更できます。ですから、メモリ(RAM)をどれだけ拡張しても、どの番地に拡張しても、このBASICは使えるようになっていきます。

ただ、1つ注意を要するのは、SAVEで格納したプログラムは、同じメモリ番地にLOADしてやらないと誤動作します。SIZEコマンドで確かめてから、SAVE、LOADすることが必要です。

**DIM** 配列は、2次元で使用メモリの高い番地から、変数領域をとっていきます。

**STOP** プログラムのデバッグには威力を発揮します。

**POKE** POKE \$E800, A \* B  
を実行すると、A \* Bを計算し、その上位8ビットを\$E800番地、下位8ビットを\$E801番地に代入します。

POKE \$0300, 10では、\$300番地に10を代入。注意を1つ、BASICのインタープリタが、\$1000番地から始まっていますから、間違ってもそこに値を代入しないこと、BASIC自体を破壊することになります。ハードの方で説明したように、書き込み禁止状態にしておくのが望ましいでしょう。

**CALL** 文 機械語(アセンブラ語)で書かれたプログラムを、サブ・プログラムと呼ぶときに使います。ただし、そのサブ・プログラムの最後は、RET(\$39)を付けます。また、スタックポインタを同じ値で戻さなければ誤動作をおこします。

**CALL** \$F58C

を実行すれば、1秒待つプログラムになります。(\$F58CはH68/TR内のROMの1つのサブルーチン)

**PEEK** 関数 これはPOKEとは逆に、指定した絶対番地の内容を変数Aに代入するものです。

A = PEEK(\$E012)

とすると、\$E012番地の内容を変数Aに代入することになりますが、実際はリレー1をOFFにするプログラムです。

(例. POKE \$E012, 00 でリレー1はONになる)

これで、一応の機能の概要がわかっていただけたと思いますが、では、1つHIT & BLOWのBASIC版を作りしたので、それを紹介しながら、構っていきましょう。

ゲームそのものは、H68/TRのアプリケーション・マニュアルにあるものと同じです。

さて、LISTを使ってBASICの補足とまいりましょう。

文番号 7 CHR\$(50C)は、画面をクリアして、カーソルを画面の左端上端に持ってくるものです。

15 文と文は、? で結ぶことができます。

20 ランダム関数は、ただRNDと指定し、RND(0)などと書かない。-32,768~32,767の間の1つの数値がとられる。このゲームは4桁を使うので、4桁未満の乱数ではだめ、よってIF文が入る。

25 乱数の絶対値をとる。

...

55 ここで初めて? = という変わった変数が現われますが、これも、このBASICの大きな特徴で、カーソルの位置指定をやり易くしています。カーソルの位置を横方向x、縦方向yとすると、H = ? を実行するとHには、x + 32(y - 1)の値が代入されます。また、逆に H = 168の時、

? = H とすると、

168 = 8 + 32(6 - 1)

ですから、カーソルは横8目、縦6つ目に移動することになります。この式を使うことで、文字を表示する際、非常に便利になります。また、このプログラム中にはありませんが@ (アットマーク) による式。たとえば、A = @ とすると、カーソルの指示した位置にある文字のASCIIコードが変数Aに代入されます。

図1 フロー

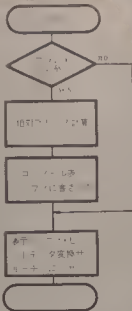


表2 \$F310からのサブルーチン

| ア    | メモ   | オペ     |
|------|------|--------|
| F310 | JSR  | \$F44C |
|      | STAA | X      |
|      | STAB | 1, X   |
|      | RTS  |        |
| F49C | TAB  |        |
|      | ANDB | #50F   |
|      | LSRA |        |
|      | LSRA |        |
|      | LSRA |        |
|      | LSRA |        |
|      | RTS  |        |

文番号 42 条件判定文では、条件が満たされなければ、次の文番号まで飛びます。

## 140 INPUT AS

A8と指示してやると、数字でなく、文字の入力を意味します。ここでは1文字ですが、1変数に2文字まで代入できます。それ以上入れようとしても、初めの2文字しか入りません。

## 160 PR CHR\$(50B). ← この点が重要

これはカーソルを1行上へ上げる文

ゲームは、リストを入れてRUN **CR** で始まります。  
数字を入れ間違えたときは、**CE** で1文字ずつ消えます。  
回数には9回です。さあ、BASICの勉強も済みました。  
少し息抜きに遊びましょう。

## 最後に

僕のマイコンも出力装置が付いて、どうにかコンピュータらしくなってきました。ただし、あくまでも「らしい」のであって、それ以上は今後の使い方にかかっているようです。今、グラフィックを使った面番着脱ゲームを考案中ですが、マイコンはゲームのためにあるのではありませんから、皆さんもゲーム以外のプログラムをどんどん作ってみて下さい。

マイコンの勉強には、どんなプログラムでも、自分で作  
ってみて失敗したらやり直す。それが大切なのです。

### 参考文献

日立マイクロコンピュータ・システム

テレビ、インターフェース、モジュール、マニュアル



## 1/0スタッフのみなさん

こんちわ

I/Oを創刊号から愛読しております。マイコンも数年間を思うと、今流行している（←後にもっとそうであると思うけど）BASICマシンは驚異的な感に及します。でも、実際BASICを動かしている人（すごい経済的余裕がある人、アタシは別）は、全体のマイコン人口から比べると、少ないのでし

なぜ、BASICがこんなに伸びたのか、  
まったのでしよう。理由はカンタン  
プログラムは楽にできるし、プログラ  
の訂正、削除、追加なんでもござれです。あの機  
械屋のプログラマーから思うと雲泥の差です。  
それと、もう1つ理由があるかと思うのです。  
番多々使われると思われるTK-80を例に  
てつものかのように、機械語からすくま  
BASICに移ってしまいました。その中間  
言語の有在のアンテナがあまりに太くとも  
発表されているソフトウェアではないのでし  
マイコンもハードウェアは比較的安  
手に入るようになりました。チップも安  
つたし、キートンも作る手間を考えると、こ  
からは、どんなソフトウェアが発表され  
べきです。(十二い、一方向論)

そこで、マイコン関係の雑誌では、番付記事が多いと思われるI/O様にお礼があるのです。アセンブラ、エディタ、ローダー、トレーサ、デバグガーついでにコンパイラ(ちよっとやりすぎかな) などぞぞ、いろいろ発表してほしいのです。

何を言いたいのか、みなさんわかってくれ

たと思いますが、これらのソフトはプログラム開発ツールとしてぜひとも備えたいものです。ただ、このようなソフトを「自分で組む」には限度があります。プロのソフト屋さんなら別でしょうが、そこで、声大に叫びたいのですが、これらのソフトを持っている人は発表したいのでしょうか？ なしかにBASICは良いけれど、アセンブリは兎を見捨てることはできない、と思うの（ふん、す）

（ふん、す）のしみです。いま、日本電機工業会（JEIC）の「ソフトウェア」をこのように読んでくれないかと思っております。この記事を載せてください。

（東京部 H.N.）

## 前 略

ちょっとしたアイデア

ミスノ」という商品名、何知ですか?「いい  
ポストカラーのような物で、いい修正液と  
同じように使われており、上から字が書けます。  
(24mlフデ、うすめ値段で480円)

これを草紙や、ICソケットの土に塗って、  
から茶色などを鉛筆、万年筆、サインペン  
で書くのです。下地が白だから鮮明です。特  
にDIの黒のICソケットのくぼみを白く塗  
るのが、最も覚えがよいようです。

(福岡市 基田彦)

ICソケット(T1型)



私が作った  
簡易エディタを使って！

最近特に思うのですが、マイコン関係の本でマイコンの選び方とか、チェックポイントとなるものが紹介されていますが、もう一つ面白さが悪い結局は、「何を選んだらよいのかさっぱりわからん」といつところに落ちつくようです。

いつの場合でも、コンピュータは、プログラムを作って命令してやらなければ、ワンともスンとも動いてくれないという、最も大切な点の説明がオロゾカにされていると思います。

こすかい。パタITEでマイコンを買ったのは良いが、アプリケーショニングプログラムをチョコチョコとやって見て、後はダルマサンよりしてきずも足も出ない。私の思いが客外多いのでなくともしょうか。(私の思うところは知らないのですが…) 8ビットの機種では、電卓出身者がどうでミニコン出身がどうといううろた惑もあるようですが、これらはすべてマイコンに關して70年代の知識、技能を恃った人達の語の談で、90年代にもおよぶ内容と、ハンドアセンブルの作業は、我々初心者にとって、最大の難関だと見えます。

さてこの辺で我が田に水を引きましょう。  
基本命令数がタノアの34種、すべて一命令で簡易アセンブラ付のLkit-16は初心者にと  
って、最も取り組みやすい機械だと思います  
Lkit-16を修繕に上げて検討中の方々、迷わ  
ずこれに決めましょう。そして、私の仲間  
になってください。(桑島 三十八郎)

# LKIT-16用

## カラー キャラクタ・ディスプレイ の 製 作

奥山昌男 (山形マイコンクラブ)

LKIT-16のメーカーから、待望の拡張基板が発売され、ワンボード・コンピュータからパーソナル・コンピュータ的使用ができるようになりました。その中でメーカー市販のテレビディスプレイはCRT方式(I/O出力)なので、自作する場合には、ハード的にコストアップになります。すでにV-RAM方式のディスプレイを持っている方は、接続方法を後述しますので、簡単にLKIT-16のグレードアップができます。

当方はV-RAM方式+7色カラーのキャラクタ・ディスプレイを製作し、フルキーを接続して、カラフルなディスプレイを楽しんでいます。

### カラー・ディスプレイの概要

図1が私の製作したカラー・ディスプレイのブロック図です。

文字数はスタンダードな横が32文字、縦が16ラインで計512キャラクタを表示できます。

キャラクタ・ジェネレータに7×9ドットのモトローラ MCM6573A を使用しているので、表示可能な文字は128種で図2の通りです。

CRTの表示方式には飛び越し走査法と順次走査法

写真1 LKIT-16と自作カラーキャラクタ・ディスプレイ



の2つの方式があります。日本のTVの標準方式は飛び越し走査法で、走査線の数は525本と決められています。したがって、毎秒30枚の画面を走査しているわけです。本キャラクタ・ディスプレイの場合はハード的にも簡単にするため、順次走査法を用いているので、毎秒60枚の画面を走査しています。現在市販のV-RAMやCRTターミナルはこの方式がほとんどです。

CRT画面上に文字を表示する場合、その対応するドットを白レベルにし、掃線期間中、映像信号を黒レベルにし、同期信号もその掃線期間中にそう入します。

図1 カラーキャラクタ・ディスプレイのブロック・ダイアグラム

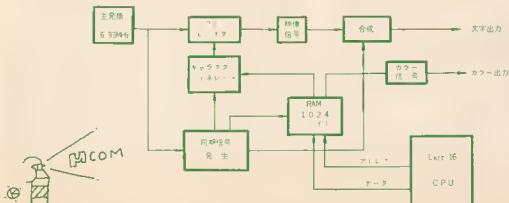




図2 MCM6573Aのキャラクタ・パターン

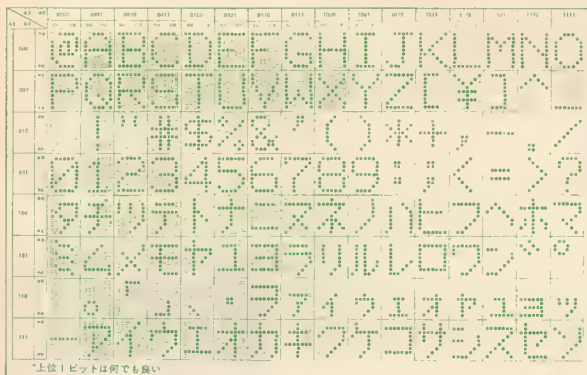
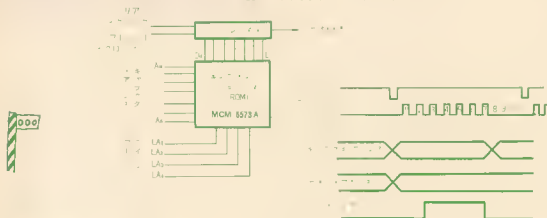


図3 キャラクタ・ジェネレータ



## 文字表示のしくみ

1文字をとのよったドットパターンで構成するかに  
より、1文字当りのドット数および1文字で使用する  
走査線数が決まっています。

今回使用した文字発生のキャラクタ・ジェネレータ  
は7×9ドットなので、1文字を表わすのに9本の走  
査線が必要です。実際は文字行間隔も3走査線使用す  
るので、本機は1文字当り、9+3で12本の走査線が

必要です。

キャラクタ・ジェネレータからビデオ信号の発生回  
路を図3に示します。キャラクタ・アドレスは文字の  
セレクトに、ラインアドレスは表示したい走査線の位  
置を指定します。

次に、キャラクタ・ジェネレータの並列出力データ  
をクロックに同期させ、シフトレジスタで、並列→直  
列変換し、水平、垂直同期信号を混合させ、ビデオ信  
号とするわけです。

シフトレジスタのスタート・タイミングはシフトロ



写真2 全キャラクタを表示させたところ

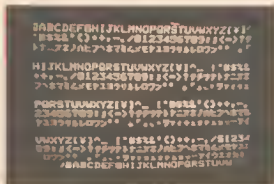
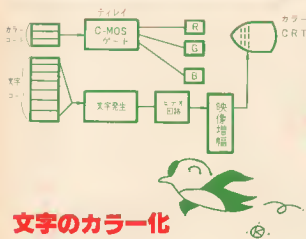


写真3 各行ごとに色を変えている



図5 RGBカラー方式キャラクタ・ディスプレイドライ回路



## 文字のカラー化

これまではモノクロと共通ですが、文字をカラー化するにはTVのRFから入力する場合は映像中間帯域の関係で不可能なので、本機はNTSC方式でなく、白黒輝度信号とカラー信号を別々に加えるRGB直接ドライブ方式を取っています。

キャラクタ出力は同期信号も含めて、第1映像増幅のベースへ入力します。

文字コードを下位7ビット、また、カラー信号は上位3ビットを使用して、V-RAMへ書き込みすれば7色の文字カラーが表示できます。

カラー信号を正論理にしてドライブすると、カラーTVを改造する必要があるのですが、私は負論理を使用し、カラーTVの改造はまったくしていません。

V-RAMのメモリ内容、カラー信号と文字色を表1に示します。

ここで問題になるのが、輝度信号とカラー信号のタイミングです。

カラーTVにはかならず、輝度信号とバースト信号のタイミングを合わせるために、輝度信号だけを遅らせるディレイ・コイルを入れて遅延させています。また、キャラクタのアクセスタイムが500nsあるので、

そのまま輝度信号、カラー信号を送ると両者間のタイミングがずれ、カラー信号が文字より先に出力され、文字と色がずれてしまうわけです。

そのため、C MOSの特性を生かしてカラー信号を遅らせカラー信号をディレイさせ、両者のタイミングを合わせています。(図5)

## 製作および調整

装置の全回路図を図6に示します。今回は44ピンのユニバーサル基板を使用していますが、カラー信号のメモリも必要ですので、試作する方には56ピンのユニバーサル基板をおすすめします。同路的にも簡単です。使用ICもスタンダード品を使用していますので入手が楽です。

キャラクタ・ジェネレータとRAMだけはICソケットを使用してください。

## 電源

今回はLKR1-16用に自作した5V 3A、-5V、+12V 1Aを使用していますが、本キャラクタ・ディスプレイの消費電流は5Vが0.5A、-5V、+12Vが0.05A位必要です。

自作の場合、できるだけ電源のリップルを取らないと映像信号を扱うので、画面にフリッカー(ハム)が入り、画質を損うので、電源2次側にはケミコン500μF位のバスキコンを追加し、また5Vラインにも適当に0.01μFのバスキコンを入れてください。

## 調整

ディスプレイ部の調整ですが、もし正常な画面が出ない時は、基本クロック発振の有無、シフト・レジスタの出力の有無、各カウンタの出力が正常に分割されているか、同期信号のチェック、キャラクタ出力と同

期信号、ブラント信号のミキシング回路をチェックします。調整用にシンクロスコープ、カウンタがあれば簡単に調整できます。

## カラー信号の加え方

カラー信号3ビットをパッパに入れて、この出力をカラーCRTのRGBドライブ回路へ入力し、そのコードに対応する色出力をコントロールします。

もし、文字と色がずれる時はCMOSのディレイ回路に入っているVR1を調整して、文字とカラー信号がずれないように調整します。

このCRTディスプレイの1ドットの長さは140nsなので、色ずれの場合はずれたドット数で色信号のずれ時間がわかりますので最適になるよう調整してください。

## LKIT-16の改造

LKIT-16の場合は未実装メモリのアクセスや、デバイスのアクセス時にデータ・アクノリッジ信号をモニタ上でチェックし、ある時間経過してもCPUに信号が応答されない時はインタラプトが起こり、バス・ロック機構が動き、プログラムの実行がその時点でロックされます。

V-RAMを接続した場合、V-RAMのアドレスがセレクトされた時はデータ・アクノリッジ信号をCPUへ返す必要があるのですが、この点がLKIT-16の拡張時のネックだったわけです。

LKIT-16を図7のように改造すれば、V-RAMアドレスがセレクトされた場合、データ・アクノリッジ信号をCPUへ送り返すので正常に働きます。

TVD-02や自作のV-RAMをすでに持っている方は

この改造だけで、LKIT-16と接続できるので各自試みてください。

メモリの増設もV-RAMと同様な方法で行ないます

## フルキーボードの接続

KEY-INでディスプレイに任意の文字を表示させる時は、モニタ・キーを使用する場合、16進のJISコードでモニタ・キーからV-RAMアドレスF000をアドレスセットして、データを書き込めば、ディスプレイ表示できるし、LEDに読み出しもできます。

また拡張SCAを増設している場合は、リスト1のプログラムを実行するとフル・キーから文字を読み込み、任意のレジスタに入力、ディスプレイできるので、応用が広がります。

私はLKIT-16+本機ディスプレイ+フルキーでカラー文字で当山形マイコン・クラブのメンバーのメモリからの住所録や電子電話簿の読み出しに使用しています。

リスト1 フルキーよりCPU R0レジスタへの読み込みプログラム

| 1600 |  | MOV  | ROX 10' |          |
|------|--|------|---------|----------|
| 1    |  | WT   | ROX 50' | モード設定    |
| 2    |  | RD   | ROX 51' | READ     |
| 3    |  | TBIT | ROX 4   | フルキーからのス |
| 4    |  | B    | *-2     | トローフ?    |
| 5    |  | WT   | ROX 51  | NO       |
| 6    |  | RD   | ROX 50' | DSR      |

フルキーからのデータ読み込み。

図7 LKIT-16の改造法

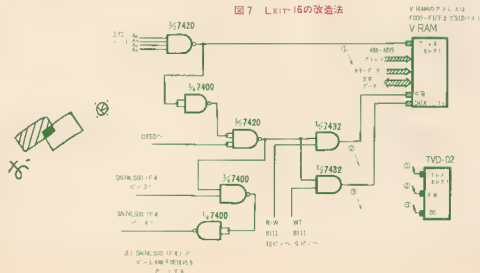
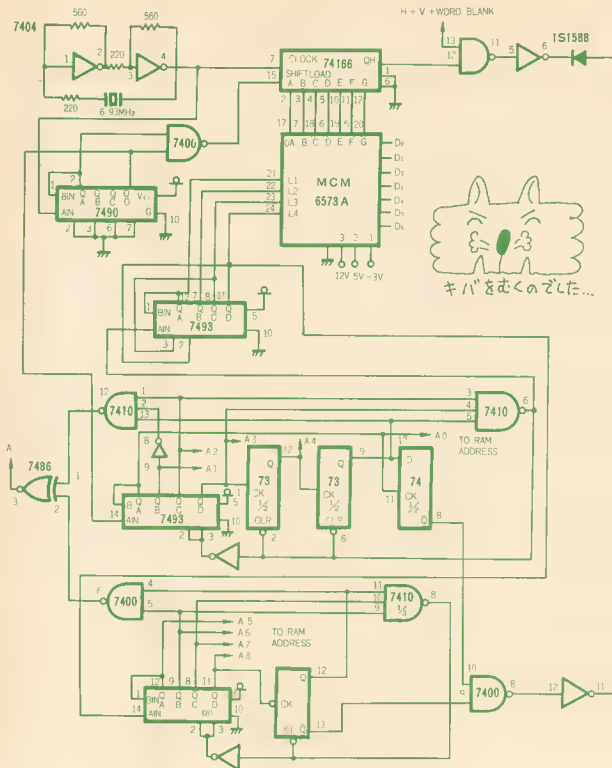
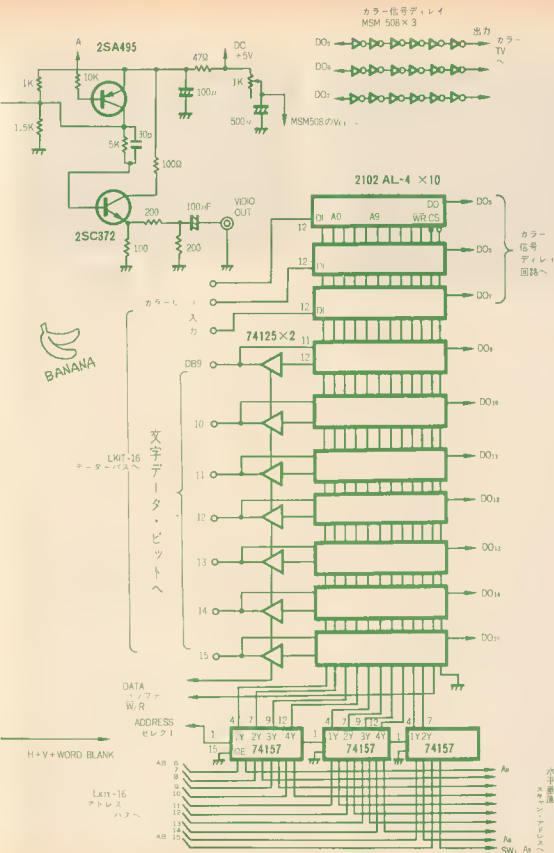


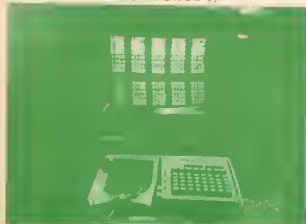
図6 カラー・キャラクタ・ディスプレイの全回路図





(主) SW1 の切り換えてへ 切り換えができる

写真4 LKIT-16とカラーキャラクタ・ディスプレイ  
(メモリ・ダンプしたところ)



-----<リスト3 LKIT-16用プログラムNo.2>-----

メモリ・ダンプ TO V RAM

①メモリ・ダンプしたいSTART番地を

STARTにセット

②メモリダンプしたいEND番地を

ENDにセット

ICカウンタにプログラムSTART番地 を書き込み

キーをオン.

V RAMエリアはF000～F1FFまでの512バイト

| アドレス | コード  | コメント            | メモ             |
|------|------|-----------------|----------------|
| 1250 | 12FF | DC 12FF         |                |
| 1    | 6202 | START1 CLEAR R2 |                |
| 2    | 822C | ST R2X2C        |                |
| 3    | CDFD | L SP+3          | スタックポインタ       |
| 4    | 08B1 | MVI R0X81       | タイマー           |
| 5    | 100C | WT R0X0C        | ウェイト 10ms      |
| 6    | 0810 | MVI R0X10       |                |
| 7    | 100F | WT R0X0F        |                |
| 8    | C90A | CLEAR L R1*+A   | F200セ          |
| 9    | C80A | L X0*+A         | F000セ          |
| A    | 0820 | MVI R0X20       | スベスフト          |
| B    | A000 | ST R0(X00)      | クリア            |
| C    | 4B01 | AI X01          | X0+1           |
| D    | 5349 | C X0R1Z         | クリア・エン         |
| E    | CFD0 | S *3            | {No}           |
| F    | C806 | L X0*+6         | F000           |
| 1260 | 8350 | ST X0X50        |                |
| 1    | CF05 | B *+5           |                |
| 2    | F200 | DC X F200       |                |
| 3    | F000 | DC X F000       |                |
| 4    |      |                 |                |
| 5    |      |                 |                |
| 6    | 6202 | CLEAR R2        |                |
| 7    | 0A01 | MVI R2X01       | 512番地にカウン      |
| 8    | B251 | ST R2X51        | 1番にセット         |
| 9    | C251 | L R2X51         |                |
| A    | 8651 | DMS X51         | 16ビット要素7       |
| B    | CF15 | B DATA          |                |
| C    | 0820 | MVI R0X20       |                |
| D    | 9050 | ST R0(X50)      |                |
| E    | C650 | IMS X50         |                |
| F    | 9050 | ST R0(X50)      |                |
| 1270 | C650 | IMS X50         |                |
| 1    | C01A | L R0X1A         | メモリアドレスをセット    |
| 2    | 8F1E | BAL DATA SHIFT  |                |
| 3    | 6101 | CLEAR R1        |                |
| 4    | 0905 | MVI R1X5        |                |
| 5    | 8151 | ST R1X51        |                |
| 6    | 0920 | MVI R1X20       |                |
| 7    | 9150 | ST R1X(50)      | スベス            |
| 8    | C650 | IMS X50         | 3ビット出力         |
| 9    | 9150 | ST R1X(50)      |                |
| A    | C650 | IMS X50         |                |
| B    | 9150 | ST R1X(50)      |                |
| C    | CFED | S START2        |                |
| D    |      |                 |                |
| E    |      |                 |                |
| F    |      |                 |                |
| 1280 | 2201 | DATA PUSH R2    |                |
| 1    | 2101 | PUSH R1         |                |
| 2    | D01A | L R0X(1A)       |                |
| 3    | BF0D | BAL DATA SHIFT  |                |
| 4    | C11A | L R1X1A         | START番地をR1にセット |

| アドレス | コード  | コメント                 | メモ            |
|------|------|----------------------|---------------|
| 5    | C21A | L R2X1B              | 512番地をR2にセット  |
| 6    | 515A | C R1R2NZ             |               |
| 7    | 972D | H                    | メモリ・ダンプ       |
| 8    | C61A | IMS X1A              |               |
| 9    | C905 | L R1*+5              |               |
| A    | 8F36 | BAL DISP             |               |
| B    | 2102 | POP R1               |               |
| C    | 2202 | POP R2               |               |
| 128D | CFEC | S *X14               |               |
| E    | 002D | DC 002D              |               |
| F    |      |                      |               |
| 1290 | CA0F | DATA SHIFT L R2*+X F | {4}           |
| 1    | C90F | L R1*+X F            | {F}           |
| 2    | 200C | SL R0                |               |
| 3    | 200C | SL R0                | 上位4ビット        |
| 4    | 200C | SL R0                | を7ビット         |
| 5    | 200C | SL R0                | 4ビット          |
| 6    | 200C | SL R0                | 4ビット          |
| 7    | 690B | AND R1R0             | 7ビット出力        |
| 8    | 200B | SR R0                |               |
| 9    | 2001 | PUSH R0              |               |
| A    | 8F0F | BAL *+X 7            | CODE EXG      |
| B    | 2002 | POP R0               |               |
| C    | 4241 | SI R21Z              |               |
| D    | CFE4 | B *-X C              | DATA SHIFT +1 |
| E    | 2003 | RET                  |               |
| F    | 0004 | DC X 0004            | カウン1増         |
| 12A0 | 000F | DC X 000F            |               |
| 1    | CC05 | CODE EXG L X1*+5     |               |
| 2    | 5C09 | A X1R1               |               |
| 3    | E900 | L R1(X1)             |               |
| 4    | 8F1C | BAL DISP             |               |
| 5    | 2003 | RET                  |               |
| 6    | 12A7 | DC X 12A7            |               |
| 7    | 0030 | DC 0                 |               |
| 8    | 0031 | DC 1                 |               |
| 9    | 0032 | DC 2                 |               |
| A    | 0033 | DC 3                 |               |
| B    | 0034 | DC 4                 |               |
| C    | 0035 | DC 5                 |               |
| D    | 0036 | DC 6                 |               |
| E    | 0037 | DC 7                 |               |
| F    | 0038 | DC 8                 |               |
| 12B0 | 0039 | DC 9                 |               |
| 1    | 0001 | DC A                 |               |
| 2    | 0002 | DC B                 |               |
| 3    | 0003 | DC C                 |               |
| 4    | 0004 | DC D                 |               |
| 5    | 0005 | DC E                 |               |
| 6    | 0006 | DC F                 |               |
| 7    |      |                      |               |
| 8    |      |                      |               |
| 9    |      |                      |               |



| アドレス | 命令        | レジスタ        | コメント |
|------|-----------|-------------|------|
| 78A  |           |             |      |
| B    |           |             |      |
| C    |           |             |      |
| D    |           |             |      |
| E    |           |             |      |
| F    |           |             |      |
| 1200 | 2011 DISP | PUSH R2     |      |
| 1    | 2011      | PUSH R0     |      |
| 2    | 6202      | CLEAR R2    |      |
| 3    | 821F      | ST R2X 0F   |      |
| 4    | 9150      | ST R1X 50   |      |
| 5    | C654      | IMS X 50    | DU*  |
| 6    | 0A01      | MV R2X      |      |
| 7    | 8F05      | BAL WAIT    |      |
| 8    | 8F08      | BAL SC1     |      |
| 9    | 2002      | POP R0      |      |
| 10   | 22 2      | POP R2      |      |
| B    | 2703      | RET         | OUT  |
| C    | C30F      | WAIT R3X 0F |      |
| D    | 5342      | CB R3R2     |      |
| E    | CFFE      | B WAIT      |      |
| F    | C 03      | BIT         |      |
| 1200 | C210      | LD R2X      |      |
| 1    | C7 4      | L R1*+4     |      |
| 2    | 5259      | C R2R1NZ    |      |
| 3    | CF03      | SC          |      |
| 4    | 27 3      | RET         |      |

| アドレス | 命令   | レジスタ   | コメント        |
|------|------|--------|-------------|
| 5    | F1FF | D      | X F1FF      |
| 6    | CB15 | SCLD L | 104+X15 F00 |
| 7    | 15   | L      | X1*+X15 F00 |
| 8    | C8FD | R *    | 3           |
| 9    | E708 | L      | R2(X1)0     |
| A    | AA 0 | ST     | R2(X1)0     |
| B    | 537H | C      | X0R0NZ      |
| C    | CF14 | B      | *+4         |
| 1200 | 4C11 | AI     | X11         |
| F    | 4801 | AI     | X01         |
| F    | CF7A | B      | * 6         |
| 12E1 | C71D | L      | R1*+X0D0    |
| 1    | CC12 | L      | X1*+XDF00   |
| 1    | A405 | SI     | R1X1        |
| 1    | 1564 | R      | X1R0NZ      |
| 1    | LF13 | B      | *+3         |
| 1    | 4CC1 | AI     | X11         |
| 1    | FFFF | B      | * 4         |
| 1    | LM 7 | L      | R1*+7       |
| 9    | R1F0 | SI     | R1X50       |
| 1    | *+5  | DMS    | X 50        |
| A    | 05   | REI    |             |
| B    | F111 | DC     | X F120      |
| F    | 1111 | R      | X F110      |
| D    | 11   | C      | X 0F20      |
| 1.EE | F1E0 | CC     | X F1E0      |

# I/Oポート

## なにわマイコンクラブ(NCC)

★我々、なにわマイコンクラブ(NCC)は、身内(同と高校生)ばかり5人のファミリー的集団で、発足したのは53年3月からです。活動(と云えるかどうか)し始めたのは53年3月からで、まだできてのホロボロですが、19才という若さでに物言わせて、ハード、ソフト(今はおもにソフト)に取り組んでいます。

★現在の活動は、BASICによるゲームプログラムの開発です。

(5人中3人がTK 80+TK 80BSを所有している)

現在開発中のプログラムは

- ・MOOゲーム(数ある)
- ・UFO撃破ゲーム
- ・迷宮ゲーム?
- ・STAR WARS

開発中のプログラムは

- ・山登り
- ・HERゲーム
- ・スロットマシン

などがあります。以上が現在の活動状況ですが、5人ともマイコンに関しては素人(TTLはまあ知っているが)なので、これからはTK 80のハード、マシン語を少しずつでいい、予定しています。

★今後の予定として、ハードは

- ・ドット・プリンタ
- ・DMA、A/Dコンバータ
- ・メモリの増設



・ハード

- ・ゲームプログラム開発
- ・データ処理プログラム

とです。

★というわけで、学生5人のメンバーですが、なにせ右も左もまじりのマイコン街道、少しでも多量知識を吸収したいと思っていますので、年令、職業に関係なく他のグループとの交流をお望みしております。また会員(と呼べるかな?)も募集しています。

★追伸

もし日本語でトイに付きたい、なつた。

●オデッセイの乱戦科(丸善館南へ、ふろ屋のとなり)の2階の階段のふもと。

●1物電機小館(丸善無縁の前面)のエスカレータののつてオデッセイ(コシヤ ホント)コーナーのリスニングルームよこの非常口付近にトイがあります。

(入会希望の方はI/O編集部へ)

村田 行



RANDOM  
Box

まず次の計算を解いてください。

A 3 9 F H + 2 C 3 2 H

どうです。解けましたか? 答えは C F D 1 H です。この問題を暗算でスラスラとできる方には、このプログラムは必要ありません。暗算でやっとなげけるという方が多いのではないのでしょうか。11ト、ハント・アセンブル時に不可欠なアドレス計算用16進4桁加算プログラムをのこしますので、TK-80をお持ちの方はどうぞお試しください。

「16進加算プログラム」リストを見て、たいへい。機械品調にHEXAD-1。また、アドレス欄にもHEXAD

という書き方をしています。(そのかわりにラベルがありません) このHEXADとは16進4桁加算のプログラム

## 16進加算プログラム 花房成臣

100

名前の1か、同時に、サブルーチン・プログラムに使用する場合のサブルーチン名も暗算地をします。このプログラムを例えは81EB番地から入ると、11HEXAD+5E81EB005E8249番地となるわけです。

使い方

| キー操作    | 表示              | コメント  |
|---------|-----------------|-------|
| 8 1 E B | 0 0 0 0 0 0 0 0 |       |
| ADDS    | 0 0 0 0 0 0 0 0 | ADDS  |
| 5 E     | 0 0 5 E         |       |
| HEXAD   | 0 0 5 E         | HEXAD |

## 加算結果

注意: 加算数(81EB)はそのまま残りますので、連続した(連続計算)が可能です。加算結果がオーバーフローした場合は、0000となります。また、数値キートン、キー以外を押すと、0000となりますので、再入力してください。

## 16進加算プログラム・リスト

| アドレス | 命令          | 説明          | コメント     |
|------|-------------|-------------|----------|
| 1    | LXI H 00 00 | HEXAD-0     | 21 00 00 |
|      | SHLD BHF1   | 3 22        | HEXAD 5E |
| 11   | SHLD BHF3   | 6 22        | HEXAD 60 |
| 12   | SHLD BHF1   | 9 2A        | HEXAD 5E |
|      | MOV C H     | C 4C        |          |
|      | MOV H L     | D 65        |          |
|      | MOV L L     | F 63        | HLE 136  |
|      | SHLD DW3    | P 22 14 83  | DW 1 480 |
|      | SHLD BHF1   | 12 2A       | HEXAD 60 |
|      | MOV C H     | 15 41       |          |
|      | MOV H L     | 16 65       |          |
|      | MOV L L     | 17 63       |          |
|      | SHLD DW3    | 18 22 16 83 |          |
|      | CALL SPCL1  | 1B 10 10 01 | ジャンプの内容表 |
|      | (CALL KLY1) | 1C 10 16 02 | ジャンプの内容表 |
|      | LPI 12      | 21 1E 12    | ジャンプの内容表 |
|      | JNZ 13      | 23 12       | HEXAD 2F |
|      | SHLD BHF3   | 26 2A       | HEXAD 60 |
|      | SHLD BHF1   | 29 22       | HEXAD 5E |
|      | JMP 12      | 2C 23       | HEXAD 0B |
| 13   | LPI 15      | 2F 1E 15    | ジャンプの内容表 |
|      | JNZ L 4     | 31 12       | HEXAD 15 |

| アドレス | 命令          | 説明          | コメント     |
|------|-------------|-------------|----------|
|      | SHLD BHF1   | 34 2A       | HEXAD 5E |
|      | ADDS        | 37 1B       |          |
|      | SHLD BHF3   | 38 2A       | HEXAD 60 |
|      | DAD H       | 3B 19       |          |
|      | JNC 11      | 3C 02       | HEXAD 5E |
|      | LXI H 00 00 | 3F 21 00 00 |          |
|      | JMP L1      | 42 03       | HEXAD 0B |
| 14   | MOV B A     | 45 47       |          |
|      | ANI 10      | 46 16 16    |          |
|      | JZ L6       | 48 0A       | HEXAD 51 |
|      | LXI H FF FF | 4B 21 FF FF | ジャンプの内容表 |
|      | JMP L1      | 4C 03       | HEXAD 0B |
| 15   | SHLD BHF3   | 51 2A       | HEXAD 60 |
|      | DAD H       | 54 29       |          |
|      | DAD H       | 55 29       |          |
|      | DAD H       | 56 29       |          |
|      | DAD H       | 57 29       |          |
|      | MOV A B     | 58 78       |          |
|      | DRA 1       | 59 85       |          |
|      | MOV L A     | 5A 87       |          |
|      | JMP L5      | 5B C3       | HEXAD 0B |
|      | BHF1        | 5C          |          |
|      | BHF1        | 60          | HEXAD 5E |

## New Products

## § BASICコンピュータ・キット §

■ COMKIT 8060は、NS社のCPU 5C/MP-IIに4K NIBL BASICを装備したアップルIIライクのBASICコンピュータ。TVを接続するだけで最少構成のシステムが実現できる。

(仕様)

▶ CPU 5C/MP-II (クロック4MHz) ▶ ROM 4K NIBL BASIC (1978年4月号p39参照) ▶ RAM 2114タイプ・スタティックRAM 1Kバイト実装 (最大12Kバイト実装可) ▶ 入力・61キーボード ASC II 配列 ▶ 出力 VHF 2ch (32Hz×16Hz) ▶ 外部記憶 オープン・カセット・インターフェイス ▶ 電源 AC100V, 15W

▶ 寸法 340×320×105mm

〈価格〉 ¥99,800 COMKIT 8060 (キット価格)

¥ 7,000 カセット・インターフェイス (別売)  
¥ 8,000 ファン (別売)

〈問い合わせ先〉 アドテック システム サイエンス㈱

〒220 横浜市西区平沼2-3-17

☎ (045) 324-1290







# 国際派のキミのための 工業英語講座

連載

MC6802  
新製品情報  
その2

高木 敦 (ESDラボラトリ)



先月号はいかがでしたか。MC6802の概要が理解していただけたと思います。さて今回も引き続き、モトロー社の Advance Information を読み進めて行きましょう。今回はブロック・ダイアグラムと内部レジスタの説明です。

## MPU REGISTERS (MPUレジスタ)

A general block diagram of the MC6802 is shown in Figure 7. As shown, the number and configuration of the registers are the same as for the MC6800. The 128 x 8 bit RAM has been added to the basic MPU. The first 32 bytes may be operated in a low power mode via a VCC standby. These 32 bytes can be retained during power up and power down conditions via the RE signal.

The MPU has three 16 bit registers and three 8 bit registers available for use by the programmer (Figure 8).

MC6802の全般的なブロック・ダイアグラムを図7に示します。図示したように、レジスタの数と構成はMC6800と同じです。128 x 8ビットのRAMが基本のMPUに加わっています。最初の32バイトはV<sub>CC</sub>スタンバイによってロー・ワ・モードで動作させることができ、RE (RAMイネーブル) 信号によって電源が入っているときも、切れたときも記憶を保持できます。

MPUには、3つの16ビット・レジスタと3つの8ビット・レジスタがあり、いずれもプログラマーが利用できます。(図8) (configuration・構成、配置・standby・待機)



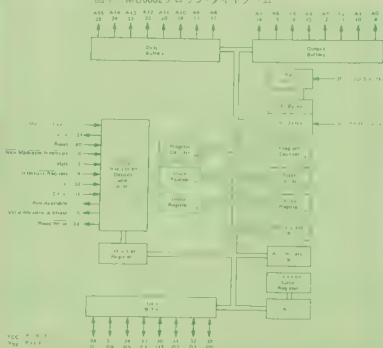
L SUFFIX  
CERAMIC PACKAGE  
QW-6 715



P SUFFIX  
PLASTIC PACKAGE  
CASE 11

## FIGURE 7-MC6802 EXPAND BLOCK DATA RAM

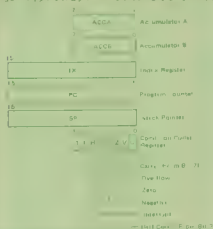
図7 MC6802ブロック・ダイアグラム



MICRO COM



FIGURE 8: PROGRAMMING THE MICROPROCESSOR  
マイクロプロセッサ・ユニットのプログラミング・モデル



**Program Counter** – The program counter is a two byte (16-bits) register that points to the current program address.

**プログラム・カウンタ** プログラム・カウンタは、2バイト(16ビット)・レジスタで現在のプログラム・アドレスを示します

**Stack Pointer** The stack pointer is a two byte register that contains the address of the next available location in an external push-down/pop-up stack. This stack is normally a random access Read/Write memory that may have any location (address) that is convenient. In those applications that require storage of information in the stack when power is lost, the stack must be non-volatile.

**スタック・ポインタ** スタック・ポインタは2バイト・レジスタであり、外部のプッシュダウン/ポップアップ・スタックで、次に利用する場所のアドレスをしまっておきます。このスタックは普通RAMで、これは都合によりこのアドレスにも置けます。電源が切れたときにもスタック中の情報を保存する必要があるような応用例では、そのスタックを不揮発性にしなくてはなりません。

(volatile: 揮発性、揮発し易い)

**Index Register** The index register is a two byte register that is used to store data or a sixteen bit memory address for the indexed mode of memory addressing.

**インデックス・レジスタ** インデックス・レジスタは2バイト・レジスタで、データをストアしたり、メモリ・アドレッシングでインデックス・モードのときに16ビットのアドレスをストアするのに使います。

**Accumulators** – The MPU contains two 8-bit accumulators that are used to hold operands and results from an arithmetic logic unit (ALU).

**アキュムレータ** MPUは2つの8ビット・アキュムレータを持っていて、オペランドを入れたり、演算部 (ALU) からの結果を入れるのに使います。

**Condition Code Register** – The condition code register indicates the results of an Arithmetic Logic Unit operation. Negative (N), Zero (Z), Overflow (V), Carry from bit 7 (C), and half carry from bit 3 (H). These bits of the Condition Code Register are used as testable conditions for the conditional branch instructions. Bit 4 is the interrupt mask bit (I). The used bits of the Condition Code Register (bits 6 and 7) are ones.

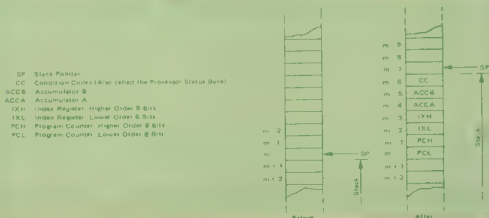
Figure 9 shows the order of saving the microprocessor status within the stack.

**コンディション・コード・レジスタ** コンディション・コード・レジスタは演算部での操作結果を示します。すなわち、負(N)、ゼロ(Z)、オーバーフロー(V)、ビット7からのキャリー(C)、ビット3からのハーフ・キャリー(H)です。コンディション・コード・レジスタの各ビットは、条件付分岐命令の条件判定に使います。ビット4はインタラプト・マスク・ビット(I)です。なお、使われていないビット6とビット7は1になっています。

マイクロプロセッサのステータス(状態)をスタックにセーブ(退避)する順番を図9に示します



FIGURE 9: SAVING THE STATUS OF THE MICROPROCESSOR IN THE STACK  
図9 スタックにマイクロ・プロセッサ・ステータスを退避させる



## I/Oバザール

## 【売 ち】

GL 2513 (5V用) ¥3.5K  
 5569 大阪府品川区津之江1-13 18 14  
 中瀬洋行

## 【売 ち】

IBMセレクトロニクスタイプライター (ト  
 ザ) TK-80 (16漢字キーボード改、ケ  
 ース、ピンコネクタ、電源付) + PSKカセ  
 トインターフェイスIC-0006 ¥100K  
 中で取りに来られる方 連絡はハガキで、  
 183 府中市白糸台 41-22 白糸台ハイ  
 ソ

小島 勉 ☎(0423)67-1696

## 【売 ち】

モニターTV、オールトランスミス  
 9インチ、調整所有、¥5K ノン  
 クロスコープ且方製、株式会社  
 ¥30Kにて急売、

173 板橋区稲荷台20-16

## 【売 ち】

NEC TK-80用ミニプロ内蔵、PD454 D  
 3を¥10K

426 静岡県藤枝市岡田山 1 14-16

池田 潔 ☎(0546)41-3435

## 【売 ち】

TK-80 (ケース、電源、トーンバス  
 CMT、RAM 1 Kbit)、TVD-01 (自作  
 ケース、電源付、ゲームプログラム)、サ  
 ンペックTVディスプレイ、2 K RAM電  
 源およびソフト付、TK 80にドッキング  
 OK、式で¥120K位で (近所ならは配  
 達します。)

490 13 愛知県中島郡平和町  
 西光坊江向 421

## 【売 ち】

藤田秀男  
 6502 TINY-BASICカセット300ボー  
 KCS ¥3.0Kにて、

274 船橋市習志野台 4-10-14

## 【売 ち】

マイックのMP-80 + アドレスの読み  
 術 (MP-80に3桁1かない) + CMTイ  
 ンターフェイス用ROM + プログラムテ  
 ープ (6種類つき) + 1 Cアンプ + MP-80  
 用の資料、以上一式 ¥40Kぐらい、くわ  
 しくは☎で

470 02 愛知県西尾市渡部二丁目  
 大平富士下小橋12

## 【売 ち】

近藤 明  
 TK-80専用TVグラフィック・ディス  
 プレイTV-32A (32x32ドット、白黒)  
 + NEC、CMTインターフェイスIC-00  
 06 (F5Kカセットテープ・インターフ  
 ェイス) + TK-80専用電源 R-15を全部で  
 ¥35Kぐらいで 連絡は☎で、

274 千葉県船橋市押引町694 (若松寮内)  
 吉田隆之

## 【売 ち】

ステーション用シンセサイザー「ACE TO  
 NE P.S-1000」529 9月購入したもの  
 で、あまり使っていないものです ¥100  
 K位で (相談により ¥90Kも可也です、  
 特にマイコン制御可能なシンセサイザ同  
 路解説用などしてくれる方) 保証書付、  
 取りにこられる方、

653 神戸市長田区東通町1-9 5 531  
 島田賢 郎 ☎(078)641 5857

## 【売 ち】

La-16用電源 ¥9Kにて、

146 大田区池上 6 42-1  
 石川元久 ☎(03)752 2009

## 【売 ち】

MK-80A (RAM 1 K漢蔵) を九十九  
 五番型ケースに入れて ¥50Kにて、自作  
 CMTインターフェイス付 (完結) 1付ノ  
 768 西川県観音寺市南町1215  
 松本芳典 ☎(0875) 7-6504

## 【売 ち】

アドテックスシステムサイエンス製TVD  
 02 TVディスプレイとキーボード (金  
 属製ケース、エンコーダー付) 一式を ¥  
 30Kにて  
 478 愛知県知多市大草字西畑 48  
 阿部伸正

## 【売 ち】

ロビン電子8 K RAMボード ¥35K、  
 SWT6 8 K BASICソフトウェア (カ  
 セットテープ) ¥3 K、

241 横浜市旭区 俣川 2-25  
 東口静夫

## 【売 ち】

H68/TRマニュアル3巻を1 Kで (送  
 別別)  
 674 明石市魚付町西岡 明石工業高等  
 専門学校学寮内  
 奥井利春

## 【売 ち】

SC/M/P KIT + テレビタイプライタ  
 20cmアルプ、TTL切替110ボー、300  
 ボー切替ケーブルコントロール付) ¥100  
 Kなるべく手渡し希望、  
 348 埼玉県羽生市小須賀926  
 早川孝史

## 【売 ち】

カセット (x-201P) を ¥20K前後で、ただし、  
 手渡し希望 平持付、  
 678 12区麻栗町赤穂郡上郡町寺井904-1  
 山本和明 ☎(07915) 2 1033

## 【売 ち】

パーソナルII + MEK 6800 D II, BAS  
 IC RUN, ユーザーメモリ 8 K、電源そ  
 の他組み込み済み、1月下旬購入近くな  
 るとどける、¥250K、

156 東京都世田谷区松原 3 7 14

## 【売 ち】

松下電器  
 マイクのMP-80電源ナンを ¥30K  
 で (完成品) 送料は当方持ち  
 960 福島県会津市大字大森17  
 本田 学

## 【売 ち】

ミニマルチ社 CL3433D ¥25K新品本  
 使用、下請し希望でカノミのマイコンプ  
 レッサーをサービスします  
 名古屋市名東区  
 早川 大郎 ☎(052)772 4521

## 【売 ち】

自作マイコン ¥COM-4 + 数字アプリー  
 タ + 16進キーボード + CMTインターフ  
 ェイス + 電源 + プログラム (オセロなど)  
 で ¥30K位、ケースはなし RAM 1.5 K  
 付、気長に待ちます、  
 302 01 茨城県北相馬郡平谷町北園  
 1411-6

佐藤 勝 ☎(02974) 8 3008

## 【売 ち】

1MSAI 8048使用のマイコンボードの資  
 料を送って下さい  
 377 群馬県渋川市大崎1541 2  
 上村 武

## 【売 ち】

TMS1121用のソケットを市価 + 送料  
 10で 5101 RAM 2個または4 K-RAM  
 1個と交換  
 654 兵庫県神戸市須磨区多月畑本戸  
 谷 3-6  
 中野 孝

## 【売 ち】

1/Oは'77年6、7、9月号を送料別、  
 定価の2割増しにて  
 603 京都府北区大宮町小野原町40番台  
 佐田 誠

## 【売 ち】

1/O込のバックナンバーならどれでも、  
 1/O割増①、H68/TRのマニュアルど  
 れでもいってから着書で送給を  
 070 北海道旭川市旭町 2 条 2 日  
 横田泰

## 【売 ち】

1/Oは '77年8月以前でTVゲームので  
 ているもの、¥0.1Kぐらいで、  
 191 東京都日野市西平山 5 16-4  
 内原 洋

## 【売 ち】

キーボード (エンコーダー付) ¥10K位  
 で 1/Oタイプ価格、  
 167 名古屋市瑞穂区浮島町 1 6 919  
 前田吉見 ☎(052)692-5491

## 【売 ち】

1/Oは創刊号〜53年2月号まで14巻を、  
 1/Oは平気で、  
 156 世田谷区上北沢 1 20-10  
 浅若啓行 ☎(03)304-3722

## 【売 ち】

SC/M/P CPUとRAM 1 KByte付き  
 で ¥1 Kで、N1 BLプログラムROM  
 を ¥1 Kで両者とも完結品で送料をもち  
 ら、☎、平持付、  
 607 京都市山科区 山科京田町43-19  
 谷 重明 ☎(075)581-7629

## ■バザール控票領

官製ハガキにこのシールを貼り①宛先 ②求む、交換の区分③署名④氏名  
 ⑤住所、⑥を記入 てください。

# 本山版 BASIC ver 7

## Palo-Altoの拡張

### ハードウェア編

Mr. Hard

私、軟弱物(ソフト)のまったくわからぬH君こと、ミスターハードです。Palo-Altoには「フローティング何か」というやつがないのだそうですね。ならば、つければいいものをソフトのS氏、いっこうにやる気配がない。それだけならまだしも、「グラフィックならコマンドをつけてもいい」などと言いだす有様でした。そこで私の登場となるのです。

例のI君(ソフト編に出てくる)が、私の所におきわすれていったV-RAMをひねくりまわしていたので、S氏のV-RAMとはちがうが、「同じようなものだ」とグラフィック・ディスプレイに改造してしまったのです。(後出の実験2)

そこまでやるなら、V-RAMは当然キャラクタ、グラフィック同時表示可能にするのは、何の手間もないのですね、ミスターハードとしましては、その手筈を記しますので写真などを参考にしながら、自分のシステムを改造してはいかがですか。ここではI君の忘れ物のサンベック8000-01というV-RAMを改造見本にします。回路図は本誌'77年12月号に掲載されています。もちろん他社のV-RAMも改造できます。S氏はTV D-02ですが、それにも私の作ったアダプタがのっており、動いています。(ソフト編の写真参照)ただ、TV D-02の回路が発表されていないので、詳細にはふれません。

したがって、以下の記事中、ことわりのない部分はサンベック8000-01の基板での話です。

#### 実験1. 256×16のグラフィック

グラフィックを表示するためのアダプタ第1号。簡

写真1 ミスターソフトのシステム。キーボードはTK-80プラグ(即ちポート)コンパチである。来月紹介予定のものです。



単明瞭、しかし最も基本的。もちろん、だれにでもできる。

#### 改造1

キャラクタ・ジェネレータ6573を基板からはずす。ハンダ付けされたICをはずすことは至難の技のひとつ、自信のある人にやってもらおう。

取りはずした跡に、KELかTIのソケットをハンダ付けしておく。このソケットに6573をのせれば、もとのまま(あたりまえ)。

#### 改造2

リフレッシュ・メモリ2101からは7ビットだけがキャラジェネに入っている。残りの1ビットの2101(基板上3A)の13ピンを6573の14ピン(本来はN.C.)に接続する。



写真2 サンバック8000-01に基本改造をほどこしたものの6573にソケットがついているのがみえる。



図1 256×16アダプタ

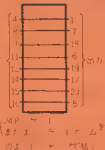
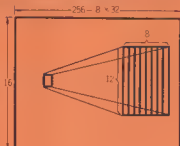


図2 256×16のTV画面



### 改造3

シフトレジスタ74LS165の11ピンがGNDされているから、このパターンをカッとし、6573の10ピン（本来N.C.）と接続する。さらに165の11ピンを1kΩでGNDしておく。（これは、グラフィックのみならず不要、こうしないと、キャラジェネをさすと縦線が32本画面に出てみっともない。）

以上の基本的な改造のうえ、図1のように24ピンのコネクタもしくはディスプレイ用プラントホームを配線し、6573のかわりにソケットに挿入する。もちろん改造3が終了したときに6573をソケットにいて、正常に動作することを確認し、電源を切ってこのアダプタと交換のうえ電源オン。

こんなものでもグラフィックとなるから面白い。ただ長いタテ線が単位となるので、何となく変なものです（そこがまた面白い……影の声）。

どうしてそのようになるかは、みなさん考えてみてください（メモリの1バイトをそのままシフトレジスタに入力するのだからあたりまえ……影の声）。

さてさて、あわせて、ここでお茶を一杯どうぞ。画面は図2のようになります。実は、このアダプタこそ、256×256のグラフィックに進む道なのです。今回はそれには触れませんが、別の機会にまた発表できたらと思っています。写真がありませんので自分でやってみてください。



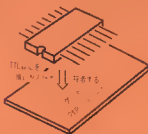
写真3 裏の様子。2本のジャンパがみえる。プルダウン抵抗はとりつけてない。



図3 なぜ64×48か？



図4 64×48アダプタ実装法



## 実験2. 64×48

なぜ64×48か？ 8ビット・マシンにおいて表わすことのできる情報は256（8ビット）であり、キャラジェネ（6573）は128キャラクタを情報としてもっています。すると、残りは128となる。128=2<sup>7</sup>だから、キャラクタひとつ分のスペース（1フレーム）に7つのドットをいれるとちょうどよいのだが、うまくいかない。フレームは図3のように、8クロック×12ラインで構成されているので、4クロック×4ラインずつに分けた6ドットとするときが、ハードウェア上最も合理的となる。1/0'77、12月号の回路をみて考えた結果がこうです。これにより必然的に表示は64×48となります。

アダプタは、これまた簡単です。私は24ピンのソケットあるいはプラントホームの上にTTLを接合剤でとりつけて配線しました。図4を参照してください。

この原理は、6573のローセレクト端子のうち、RS2、RS3は、ここだけみれば、3進カウンタでドライブされているので、これをフレーム内の縦3ドットの切り換え信号とします。これにより、自動的に4ラ

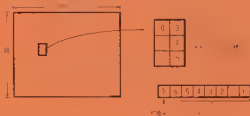
写真4 メモリの初期値の表示。  
キャラクタとグラフィックが混在している。



写真5 乱数によるパターン例。OM-2のシャッター幕が  
この例だ。



図6 64×48



インずつとなります。横方向の切換えは、セレクトの出力をシフトレジスタの前半分の4ビット、後半分の4ビットそれぞれに同時に加えることによります。

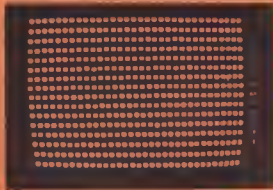
ここではセレクトに74LS153を使っていますが、私の試作機では、手元に153がなかったため157をよたつ使って作りました（もったいない…影の声）。

原理さえ理解すれば、何の造作もなくできてしまうので、手持のパーツを使ってください、といっても153を使うことが一番簡単と思われる。（図5）

フレーム内のドットとメモリのビットとの対応は、153の入力とメモリの対応で決まるので、これも自由に変更できます。

図6のビットパターンは、BASICでPLOT、PICKを処理する上で、プログラムを容易にするためにMSBでキャラクタとグラフィックの切り換えを意図して

写真5 グラフィックの例。このパターンはLKIT-16  
による自製プログラムでできたもの。

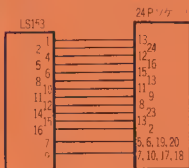


LKIT-16、自製プログラム

```

a      L, SP  *+7
a+1    BAL  *+0
a+2    DC   a
  
```

図5 64×48アダプタ配線図



決定しています。（次節参照）

もちろん、ビット・パターンを横一列にならべる方法が最上ですが、Video-RAMの大改造を必要とするので現実的ではありません。ただし、新たにディスプレイを設計するのなら話は別です。

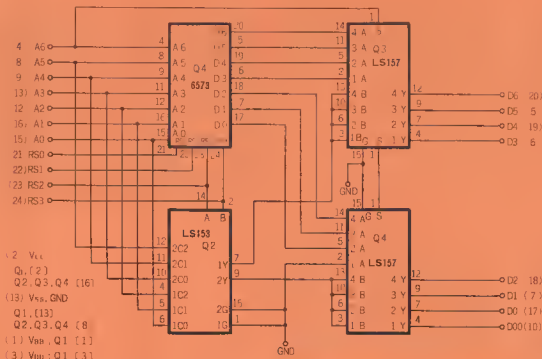
実際に私が弾かめたのではないのですが、タンディラジオシャックのTRS-80のグラフィックは、私がここに書く方法と同じらしいのです。TRS-80をもっている方、ぜひ教えてください。

### 実験3. 64×48+キャラクタ

いよいよ本題の『キャララグラ』の登場ですが、もう、ここまで書いてしまうと先を読まなくてもできますね。読んでいるのは、キャラクタとグラフィックの切り換えをどうするかという点だけです。6573なるキャラジェネを100%生かして使うには、ビット・パターンの最上位の2ビットを11または、00としたときにグラフィックとなるように、切り換え信号を作るのが良いのですが、S氏と打ち合わせたところ、2ビットの操作がめんどうだといっているやがるのです。

おまけにS氏はもちろん、I君、B君、そして私もかなぞ使いませんので、それならMSBのみを切換

図7 64×48-キャラグラフィアダプタ。( )内は28Pのピンナンバー、( )内は各ICのピンナンバー。



信号に使うという話は、1分とたたないうちに決まりました。

他のCPUでビット操作の容易なものを（例えばパナファコンのMN-1610；LKIT-16）でドライブするときは、最上位2ビットをつかって切り換え信号をつくれれば、6573の機能は生きます。

回路は図7のようになります。私は40×70ほどのユニバーサル・ボードに組みました。かなり細かい配線をしなければならぬし、アダプタとV-RAMとの接続方法もこれまでのアダプタよりシビアになります。接合剤をうまく使えば、IC4つおんの大きさにすることが出来ます。

ハード屋のミスター・ハードとしても、このこまかいやつを手で配線するのは3つもつくと、もうイヤダとなげ出したくります。お金を出すから誰が作ってください。私用のアダプタを、だってなぜか、私の手元には、ひとつもないのだから。

## 実験 終わって夜が明けて

ここで作れば、もう、グラフィックTiny Basic本山版ver. 7は、キーボードをのぞけば、すぐ走る。来月は、そのキーボードを紹介するから、それまでにV-RAMを改造しよう。もう1度叫びたい。「私に誰かアダプタを使ってくれ!!」じゃ、また、来月!

写真7 図7の回路のアダプタをとりつけた状態。



## 参考文献

- 1) I/O '77, 12, p. 49-55, 工学社, 1977
- 2) 最新TTL IC規格表 '77, CQ出版社, 1977
- 3) モトローラ・データ・シート, 「MCM6573」



# ソフトウェア編

Mr. Soft

発端は、髭と長髪で東海マイクロ界で有名(?)な I 君が、粵明市の友人から借りてきたという Palo Alto のテープを持って、私の所へ現われたことより始まります。当時、私は TK-80 にメモリを 4 K ばかりと V-RAM だけというシステムで、遊んでおりました。

『こんなものがあるんだけど、お宅のシステムで動くかな? メモリと V-RAM は、ちょうどいいんだけど。』

『何とかするから コピーさせてちょうだい!』

『キーボードもないのにどうするの?』

『キーボードぐらい自分で組立てて、ソフトウェア・スキャンで、工作は H 君にたのめるし……』といった調子で、しるし I 君から無理にテープを取りあげて、本山版 BASIC ver. 1 と名づけ、さっそくインタープリタの移植を始めた。

TK-80 の 25 個のキーを 3 段にシフトして、フルキーボードにした ver. 2、借りもののキーボードを利用した ver. 4、オセロで遊んだ ver. 5 等々、小改良も含めて改定は 20 版を越え、半年たった今、予定の“ソフトウェア・スキャンによる TK-80 のキーボードとプラグ・コンパチブルのフルキーボード”による最大の版 ver. 7 が完成したので I/O 誌にお願いして、発表させていただくことになりました。

詳しいことは、ハード編の H 君にまかせることとして、必要なハードを書きますと、

## ① TK-80

② V-RAM (アドテックの TVD-02 または、サンベックの 8000-01 を改造したもの)  
アドレス… \$FE00 ~ \$FFFF

③ メモリ 最低 \$8000 ~ \$FFFF の 4 K。記事の中で

図 1 システム構成

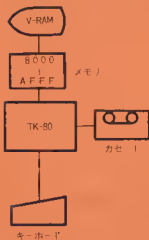
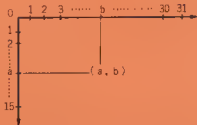


図 2 キャラクタ・モード



は \$8000 ~ \$FFFF の 8 K を想定。

④ キーボード 次号で紹介予定のソフトウェア・スキャンによる TK-80 の 16 進キーボードとコンパチブルなフルキーボード。

⑤ FSK カセット・インターフェイス となっています。(図 1)

ソフト担当の私としては、今回は連載第 0 回ということで、Palo-Alto に私が追加したコマンドの一覧表に、簡単な説明をつけ加えたものを書きます。次回以降、順を追って追加コマンド用の BASIC の改造部分の機械語と説明が私の担当です。

新しいコマンドの使い方については、例を挙げて B 君が説明してくれることになっています。

## Ⅰ. キャラクタ・モード

16 行×32 字の画面の位置を、図 2 のように、1 組の数字であらわします。すなわち、a 行目、b 字目の位置は (a, b) というわけです。

## PEEK(a, b) (省略形 PE.)

(a, b) の値を関数の値として持ち帰る。

## POKE(a, b) (PO.)

ver. 7 の特徴あるコマンドの 1 つで次の 4 通りの使い方ができます

### ① PO.(a, b) X

X の値を (a, b) に代入する。

### ② PO.(a, b) \* X

X の値を 10 進 5 桁になおして、(a, b) より表示。

### ③ PO.(a, b) \*.....\*

\*.....\* の内容をそのまま (a, b) より表示。

### ④ PO.(a, b) \$h1h2

h1h2 は 16 進 2 桁、すなわち 1 バイト分の数で、(a, b) に h1h2 を代入します。すなわち、



写真1 システムの全景。ハーフトーンを利用して、オセロゲームを行なっているところです。プログラムはBASICで書かれています。



写真2 TVD-02をグラフィックおよびハーフトーンが出るように改造したもの。



ASCIIコードで直接画面へ書き込むことができます。

さらに①-④までを、PRINT文と同じ要領で継続が可能です。(カット写真参照)

## CURS(a,b) [CU.]

カーソルを(a, b)へ持って行きます。このとき、前のカーソルは消され、また新しいカーソルは表示しません。

## ERASE(a) [E.(a)] またはERASE(E.)

①E.と引数をつけないか、またはE.(16)で全画面を消去します。

②E.(a) 0 ≤ a ≤ 15  
第a行を消去します。

## II. グラフィック・モード

今回のハード編の話題になっているように、V-RAMの改造によって可能になったモードで、キャラクタ・モードと同じように、画面を48×64の画素に分割(1つのキャラクタは6分割されます。)、その位置を(a, b)とあらわします。(図3)

## PICK(a,b) [PL.]

グラフィック・モードのPEEK命令と思えます。画面の(a, b)が黒(普通のディスプレイ

図3 グラフィック・モード

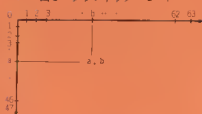


図4 キャラクタと画素との対応



キャラクタ・モード      グラフィック・モード

写真3 POKEとPAUSEに注意。

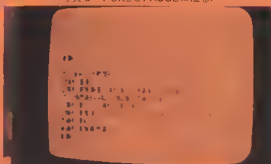
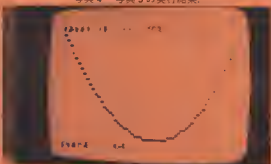


写真4 写真3の実行結果。



レイでは、明るい点)なら"1",白なら"0"を値とする函数。ただし、その画素を含むメモリが、キャラクタ・モードであれば値は255(FF)となります。(図4)

## PLOT(a,b) [PL.]

POKEに当るコマンドで、やはり次の3つの使い方があります。

- ①PLOT(a, b)  
画素(a, b)を黒くします。
- ②PLOT(a, b) [PL. B]  
画素(a, b)を白くします。
- ③PLOTI(a, b) [PL. I]

写真5 グラフィック・モードとキャラクタ・モードの  
関連に注意。

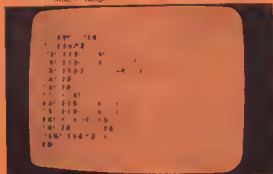


写真6 写真5の実行結果。  
オリンパスOM-2のシャッター17



図5 メモリ・マップ

|      |                                                      |      |                       |
|------|------------------------------------------------------|------|-----------------------|
| 8000 | PICK, PLOT<br>BELL, PEEK<br>FNC, POKE<br>ERASE, CALL | 8C50 | KEY<br>PAUSE          |
| 81FF |                                                      | 8C9C |                       |
| 8200 | 画面出力<br>ルーチン                                         | 8C9D |                       |
| 82AF |                                                      | 8D15 |                       |
| 82B0 | CURS<br>RMZ                                          | 8D16 | プログラム<br>領域           |
| 82DF |                                                      |      |                       |
| 82E0 | RND<br>POKE(続き)                                      | 9FFF | FNC文<br>飛び先           |
| 83BF |                                                      | A09F |                       |
| 8400 |                                                      | A0AC | CALL文<br>飛び先          |
| 8BD7 |                                                      | AE5F |                       |
| 8BD8 | 命令および<br>関数<br>テーブル                                  | AE60 | キーボード<br>サービス<br>ルーチン |
| 8C5F |                                                      | AFFF |                       |

## PAUSE X (PA.)

この文を実行すると、画面左下に

PAUSE X1X2X3X4X5

とXの値を10進5桁で表示し、プログラムの実行を中止します。キーボードより[R]の入力があれば、次の文の実行に移ります。

## KEY (K.)

キー入力の有無をしらべるルーチンで、キー入力があればその値を、なければ、16進FFを関数の値として持ち帰ります。

画素(a, b)を白黒反転します。

このコマンド実行時、対応するメモリがグラフィック・モードでない場合は、そこをグラフィック・モードの全白に切り換えて文を実行します。

## EG

グラフィック・モードでの全画面消去です。

注 キャラクタ・モードのスペースは\$20ですが、グラフィック・モードでは\$80です。

## 類III. 音に関するもの 8255のPC1に出力します

### BELL(k) [B(k)] またはBELL (B.)

引数なしで、中央音を、引数付きの場合は、その引数に反比例する音を100msほど発生します。

### SON, SOFF

各々、キーボードからの入力のたびに確認のために音を出すか、出さないかのスイッチです。

## 類IV. その他

### FNC0( ) [FN, 0]

### FNC9( ) [FN, 9]

ユーザーが機械語で書いたサブルーチンへのジャンプ命令です。0~9の数によって飛び先は指定されています。引数の値をサブルーチンに持って行って、さらに数値を持ち帰ることできるよう、関数として定義されています。

### CALL(a) [CA.]

引数aの値によって決まる番地へのサブルーチン・コール命令です。

## RMZ( ) (RM.)

引数の値だけ乱数発生ルーチン呼び出すもので、乱数の初期化にもちいます。

## 番外. RND (X)

Palo-Altoでは、メモリの内容を乱数表として用いていましたが、64ビットの疑似乱数を発生させるようにしました。函数として、Xの値が止ならばPalo-Altoと同機能ですが、0または負のときは、0~32767の間の乱数を発生します。

以上のように、14ものコマンドを追加しましたので、一度にすべてを発表というわけにいきませんから、連載の形で発表して行きます。基本となったインタープリタは某社のものですが、一般的ではありませんので、東大版の解説付の代表として、「bit」増刊号インタープリタに対する、フランチ先などのアドレスを併記します。で、読者自身のインタープリタを私の記事を参考として、改造が可能だと思います。図5に現在のメモリ・マップを示します。割線部がもとのインター

プリタです。

最後になりましたが、Palo Alto版を開発したWang博士、それを掲載したD. D. J., さらに日本への紹介、普及に努められた東大の方々、インタープリタのコピーを黙認してくださった某社およびI君に感謝いたします。

## □参考文献

- 1) TK 80ユーザーズ・マニュアル  
NEC IEM 560C AUG. -19-77
- 2)  $\mu$ COM-80Fユーザーズ・マニュアル  
NEC IEM 585A SEP. 24 77
- 3) Li-Chen Wang: Palo-Alto Tiny BASIC.  
dr. dobb's Journal of COMPUTER Calligraphy & Orthodontia vol.1 No.5 1976
- 4) 小野芳彦「Tiny BASICインタープリター」  
bit 増刊号 1978年2月 p.277
- 5) 伊藤 誠「Palo-Alto版Tiny BASICのすべて」  
インターフェース 1978年4月号

## New Products

§ ソード電算機  
§ ホーム・コンピュータ発売 §

ホーム・コンピュータとしては、現在PET, TRS-80, APPLEIIなどに人気を集めている親があるが、このたびソード電算機ではBASIC内蔵のホーム・コンピュータM100を発表した。

▶ M100はCPUにZ-80を採用。BASIC内蔵で、RAMは16Kバイト。オーディオ・カセット2台接続可能。特長は8ビットのA/D変換器。デジタルI/O各8本。モザム・インターフェイス、タイマ、時計。S-100バスが使えることなどで、従来の機種に比べて付加機能が多く、研究開発用ばかりでなく、家庭、工場などの現場でも充分活用可能と思われる。事実、すでにビニールハウスの温度制御用に交注しているという。

▶ ホビーストとしては、標準装備のジョイスティックはゲーム用として活用できる。

▶ 本体19万9千円、モニタTV込みでも29万9千円というのは、TRS-80, PETを考慮した価格と思われるが、M100はコスト・パフォーマンス上充分競えると思われる。

▶ ソードでは海外への輸出も考えており、米国からの一時的な輸出を許してきたのが同のホーム・コンピュータ界隈もここにきて、ようやく反撃に転じてきたといえよう。

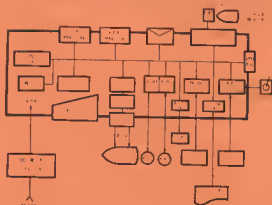
(仕様)

## ●標準構成

- CPUにZ-80採用
- メモリ16KバイトからMAX 32Kバイト(内蔵可)
- 64×24文字(白黒)モニタ・テレビ・コン

- オーディオ・カセット2台接続可能
- アナログ信号変換器2台内蔵(8bit ADC)
- デジタル入出力各8本内蔵(リレードライバ、フォトカプラ付)
- オプション
  - 5V 4A電源
  - 白黒モニタTV、カラ モニタTV
  - カラーグラフィックコントローラ (M100, M200 共用可)
  - フルボードおよびベリシックキ、ボード
  - ジャーナルプリンタ・インターフェイス内蔵
  - S100バス使用可能
  - モザムクプラ・インターフェイス(RS232C)
  - 時計
  - オーディオ・カセット拡張
  - S100バス用拡張筐体
  - AC100V電源コンセント・コントローラ

M-100 ブロック図



ジョイスティックレバ 1本内蔵  
2オクタブ内蔵のスピーカ内蔵

価格)

- 本体 ¥199,000
- オプション(モニタTV、カセット・レコーダ)
- 込み ¥299,000
- 問い合せ先)
- (株) ソード電算機システム
- 〒124 東京都品川区内新小岩4-42-12
- 税別第2ビル 電話(03)696-6611

家庭用TVで使える?

64カラム×16ライン×4ページ

全回路図  
一挙大公開

# キャラクタディスプレイ ICT-4096ターミナル

(インターフェース技術部)

## ディスプレイについて

### 飛越走査

テレビで絵を走査するには、図1のように走査面の左上から始まり、下にわずかに傾きながら実線のように右に進んで右端に達し、そこから、行きの数倍の速さで左端に戻り、最初の出発点のすぐ下から次の走査線が始まります。

そして、順次に走査が進んで最下端中央点まできたとき急速に最上端の中央点まで戻り、前の走査線の間を点線で示したように走査する方式がとられ、2回画面を荒く走査して、完全に一つの画面を走査し終わります。

このような走査方式を飛越走査(interlaced scanning)と呼び画面を荒く1回走査することをフィールド(field)走査といい、2回のフィールド走査で完全に1画面とします。

1秒間に完全な画面を送る枚数をフレーム周波数(frame frequency)といい、わが国では現在30Hzで行なわれています。これに対して、フィールド周波数は60Hzということになり、飛越走査によって、ほとんどちらつきの感じられない、高密度の画像となります。

飛越走査を行なうには水平同期信号・垂直同期信号

・帰線信号の他に、垂直同期信号の前後に等化パルス・垂直同期信号中に切れ込みパルスなどが必要となり、その周波数・パルス幅・移相などの関係が非常に複雑になります。

そこで、一般のCRTターミナル(ビデオ・ディスプレイ)の場合は、簡略化するために飛越走査を行わずに、水平・垂直同期信号・帰線信号のみを合成し、走査線に關しても本来は1フィールドあたり262.5本(525/2本)のところを、だいたい260本前後としています。(262.5本より多い場合は画面の上下で走査線の間隔がつまり、少ない場合はのびることになりますが、それでも画像は出ます。しかし、そのために水平・垂直同期の周波数がずれるなどして、安定度・画面のバランスなどが悪くなります。)

したがって各フィールドの移相が同じですので、偶数フィールドの走査線は奇数フィールドの走査線の上に重なってしまうので、垂直方向の密度が荒くなります。

これに対して、飛越走査方式では、位相をずらして

図1 飛越走査

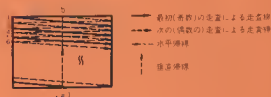


図2

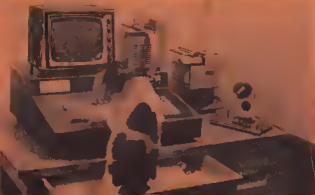
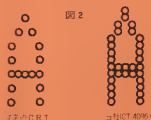
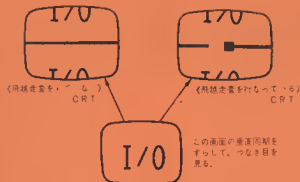




図 3



偶数フィールドの走査線が、奇数フィールドの走査線のすき間にはいるようにしているので、走査線が262.5×2=525本となり、上記のものに比較して、垂直方向の密度が倍になり、非常に鮮明な画像になります。

### 《飛越走査を行なっているかどうかの

#### 簡単な見分け方》

まずテレビの垂直同期のつまみをわずかに回して映像が上から下へゆるやかに移動するようにしておき、次に輝度およびコントラストのつまみを調整して、画面のつなぎ目の部分がよく見えるようにします。

ために、一般のテレビ放送を受信して、同様に操作してみてください。図3のようになるはずですが、

### 各信号のタイミング

1つのフィールドが終わりに近づくと、等化パルスがやってくる。等化パルスは水平同期信号の尾のパルス幅を有し、2倍の周期でやってくる。すると、バカ者のテレビは、今まできちんと足なみそろえて水平同期パルスを出してきたはずなのに、ここで、ガクッ!と歩調を狂わされてしまう。

そこへ垂直同期パルスが来て、その後またタメ押しの等化パルスがやってくるので、次のフィールドへはいつ、水平同期パルスを出した時には、いつのまにか、前のフィールドに対して180度移相がずれているというわけなのです。

これによって、このフィールドの走査線は、前のフィールドの走査線のすき間にちょうどはいるという事になるのです。

メタシ メタシ。

バカなのはテレビ者だけ。1回のみならず、各フィールドごとに、このようにだまされているのだから。

ICT-4096では、これらの信号をいっきに変換することなく、心臓部にX'101を使用し、その誤差10°のオーダーで周波数およびパルス幅とも、わが国のTV規格を完璧に備えています。

図 4

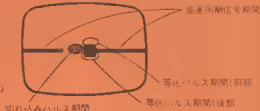
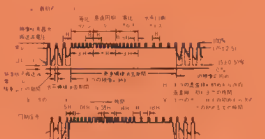


図 5



図 6  
この図は、水平同期パルスと垂直同期パルスのタイミングを示している。

図 6



水平同期信号期間  
切れ込みパルス期間  
等化パルス期間(前部)  
等化パルス期間(後部)

したがって、どんなTVでも(早朝に町内を散歩すると、白黒テレビの1台や2台必ず捨ててあります。小型のテレビを見つけたらぜひ、拾ってきましょう。たいてい。ブラウン管を交換するだけ(3000~4000円)で、新品同様になりますよ。 絶で、確実に同期がかり、画面位置は調整でブラウン管のまん中となります。

「同期周波数微調整」・「画面位置調整」なんて、ヤボな調整箇所はないのです。

どうです、すごいでしょ。

### 64文字×16行×4頁のスクローリング

つまり、64文字×64行分のデータが記憶できるので、





画面のオーバなどなんのその。気にせず、どんどん打ち出してください。(画面の右下まで行くと、自動的に1行分シフトアップします。)

そして、打ち出し終わってから、おもむろに、タイプライタのプラテンを回すように、画面を巻き戻して(キーボードの←キーを押す)画面外にかくれていたデータをじっくり見なおします。

次に、←キーを押して画面を一行ずつUPしていき、最後までデバックします。

頁はすべてリング状につながっていますので、行き止まりになる事はありません。上にも下にもくるとマニュアルで回す事ができます。

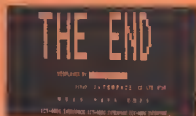
カーソルも、↑、↓、←、→、などのキーを押せば自由に移動できます。これらのキーはすべてオートディレイ・リピート機能によって、押し続けるとびと呼吸おいてから、連続動作します。

すべての動作機能はハードで構成されているので、制御用のソフトはいっさい不要です。

カーソルはソフトで制御する事もできます。そのうえカーソルブリンク機能(点滅)付なので、どの位置にカーソルがあってもすぐに見つける事ができます。カーソル移動中(文字出力中)は、ちらつかないようにブリンクが停止します。

**HILITE** キーを併用すれば、任意の文字の白黒反

写真22



転ができます。(ソフト制御も可能)。全画面を白黒反転(白バックに黒文字)したい場合は、リバースモードにします。リバースモードにおいてもハイライティング機能が使えます。

### キーボード

キーボードの16進キーは、電卓と同配列の上にA～Fまでのキーが付加されているので、電卓を便いなれている方は、マシン語でのプログラムを非常に早く打ち込む事ができます。(1時間に3KB程度は可能です。)

キートップも、機能別に色分けをし、使用ひん度の高いものに関しては、大きなキートップを使用しています。文字はすべて彫刻で、特にJIS仕様のキーボードでは、英語モードの文字を太く大きく彫刻して、JISキーボードの見た目を改良しました。

マイコンだよ おがさん!

### 中京リポート

- タケイ無線 3月のUSA品間で小物を沢山買入れて来た。オーケーツールなどはすぐ売ってしまった様子。Appleも買って来たが、まだ店頭には売れない。液晶クロック・キート¥4,500、¥7,800の2つはおもしろい。
- ラジオセンター工器具 1台は欲しい万力が¥5,800、シャコ万力は¥780から。
- 2下の平石通画ではK55Lのエッジ・コネクタが定価の半で売られている。
- 地下鉄赤坂・五反田近くの萩原電機(オギワラデンキ、コスモス名古屋のひとつであった)にはミニコンの端末、ファクシミリなどが置

いてある。ただし販売員はいません、全員技術屋さんです。

- カトー無線 新設部にて大々的にがんばっている。4月22、23の両日の内覧会ではサンヘック8000 TKにA S R-33が壊れ、スターレックが走っていた。2114 ¥2,800が月世商品、(パーツ、マイコンは5F)。
- スリーエス名古屋では、2102-AL4 ¥300!!

「まだまだ安くなるでしょう」とのこと。ちっくしょう、ほかの店で¥350で買ってきたところなのだ!!

- 例によって名古屋マイコン女性情報会情報

トヨムラのケイコちゃんは、「10」などにのったお版にいけないワ」とトヨムラをやめ証券会社に移ってしまった。それに、それに、マイコンシステムセンターのモリタさんは、サントややじい、男と結婚してしまったのだ。でもおわりにひとり来たもんな。

●器楽魂(いっつ、まっつ)で生タマゴ1個¥150。このタマゴ、どれをとっても炭身が、2つ以上入っている。もしも3つは入っていたら、店長にみせよう。食事代金がたになる。ホントだよ

(リポーターは料理 儀)

# Z-80

## マイクロコンピュータの製作④

### Z-80の命令 演算

東 光一

#### ◆算術演算と論理演算

コンピュータもロード命令のように、ただデータをやりとりしているだけでは、ゲームを楽しむことも、機械を制御することもできません。そのため、算術演算、論理演算を使って必要とするデータだけを作り出します。算術演算とは、足し算、引き算、コンペア、インクリメント、デクリメントなどです。論理演算では、アンド、オア、エクスクルーシブ・オアです。

#### ◆算術演算

足し算、引き算には大きく分けて2種類あります。これは、下の桁からのくり上りを演算時にいっしょに演算するか、くり上りを無視するかです。このくり上りを記憶するものとして、キャリーフラグがあります。それでは、実際の演算を行なってこのフラグの働きを調べてみましょう。図1を見てください。

この2つの演算の答が同じだと信じられますか。このままではわかりにくいので、8ビットの2進数の足し算にしてみました(図2)。

どうです。答の下位8ビットは両方とも同じ値になっています。しかし、9ビット目は違っています。ここで、演算の結果をしまっておくレジスタが8ビットだと結果は両者とも同じになっています。しかし、マイコンはえらいもので、9ビット目を専門に記憶する

図1

|       |      |
|-------|------|
| 67H   | 07H  |
| + C8H | +28H |
| 12FH  | 2FH  |



図2

|           |   |           |   |
|-----------|---|-----------|---|
| 01100111  | 8 | 00000111  | 8 |
| +11001000 | 8 | +00101000 | 8 |
| 10010111  | 8 | 00101111  | 8 |

部分があります。これがキャリーフラグです。もう一度、図2を見てください。演算後、9ビット目がキャリーフラグに入っていれば、演算結果に256 Dを加えればよいのです。

それでは次に、減算の場合について調べてみましょう。

図3を見てください。9ビット目の答が正のときは0、答が負のときは1になっています。正の場合は答はそのままで正しい値ですが、負のときは話がちがってきます。答が負でも、その桁の上はまだ桁があれば、この9ビット目もそのとき、いっしょに引いてやればよいのです(図4)。

実際の命令ではこの9ビット目はキャリーフラグに記憶されているので、次の演算のときに、キャリーもいっしょに引く減算を行なえばよいのです。(減算の場合は、キャリーというより、ボローと呼ばれています。)上にまだ、桁があるときはよいのですが、ないとき

図3

|           |           |
|-----------|-----------|
| 68H       | 27H       |
| -39H      | -48H      |
| 2FH       | -21H      |
| 01101000  | 00100111  |
| -00111001 | -01001000 |
| 00010111  | 11101111  |

図4

|     |           |
|-----|-----------|
| 123 | 00100011  |
| -47 | -01000111 |
| DC  | 11011100  |
|     | 00000001  |
|     | -00000000 |
|     | -00000001 |
|     | 00000000  |

図5

37H → 00110111B  
37H + 11001001B

図8

元の数A 01010101  
最上位ビットを1にする場合  
A 01010101  
B 10000000  
ABのORを取る  
答 11010111  
最上位ビットを1にする場合  
A 01101010  
B 01111111  
ABのANDを取る  
答 01010101  
24ビットを反転させ  
174をそのままにする  
A 01101010  
B 11100000  
ABのXORを取る  
答 10101010

はどうなるでしょうか。ここで2の補数と言う考え方が出てきます。

## ●2の補数

皆さん、足し算、引き算を行なうとき、負の数はどのように表わしますか。数字の前に（マイナス符号）を付けますね。では、コンピュータはどのような符号を使うのでしょうか。符号と言っても、2進数を使っているのです。1と0しかありません。そこで、演算に使う2進数の最上位ビットを符号として使います。0が正を表わし、1が負を表わします。

それでは、その方法によって数を書いてみます。

(図5)

2つの数は、もともと符号が違っているだけですが、2進数に直したものを比べると、まるで関係のない数のようです。これは、負の数が符号だけ変えるのではなく、2の補数になっているからです。2の補数の作り方は次のようになります。(図6)

これで、2の補数ができたわけです。この数を使うと足し算、引き算が簡単に行なえます。例を次にあげます。(図7)

2の補数を使えば、さっきの減算で答がマイナスになる場合が解決します。絶対値が小さいときは、答の2の補数を作ればよいのです。

その他にも算術演算にはインクリメント、デクリメントがあります。それぞれレジスタの内容に+1、-1することです。また、コンペアという命令もあります。これは、レジスタAとデータを比較するものです。レジスタAの内容とデータの内容が等しいとZフラグがセットされます。

図6

45Hを作るとき  
作りたい数の絶対値の数を2進数になおす  
01000101B  
その数のNOT 否定を作る。  
10111010B  
新しくできた数に1Bを加える  
10111011B

図7

45H 01000101  
+ 77H +10001001  
32H 11001110  
  
45H 01000101  
+ 77H +01110111  
32H 11001110

表1 74181

|     | A  | B  | C  | D  | E  | H  | L  | H  | IX+di | IV+di | n |
|-----|----|----|----|----|----|----|----|----|-------|-------|---|
| ADD | 87 | 80 | 81 | 51 | 83 | 84 | 85 | 86 | 1E    | 1F    | 0 |
| ADC | BF | 89 | 89 | 5A | 8B | 8C | 8D | 8E | CE    | CF    | 1 |
| SUB | 97 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 05    | 06    | 0 |
| SBC | 9F | 98 | 99 | 9A | 9B | 9C | 9D | 9E | DE    | DF    | 1 |
| AND | A7 | A0 | A1 | A2 | A3 | A4 | A5 | A6 | E6    | E7    | 0 |
| XOR | B7 | B0 | B1 | B2 | B3 | B4 | B5 | B6 | EE    | EF    | 1 |
| OR  | B7 | B0 | B1 | B2 | B3 | B4 | B5 | B6 | FE    | FF    | 1 |
| CP  | BF | B8 | B9 | BA | BB | BC | BD | BE | FE    | FF    | 1 |
| INC | 3C | 04 | 0C | 14 | 1C | 2C | 3C | 34 |       |       |   |
| DEC | 3D | 05 | 0D | 15 | 1D | 25 | 3D | 35 |       |       |   |

B: 10ビット、L: 16ビット

## ●論理演算

この演算には、アンド、オア、エクスクルーシブオアがあります。これらの演算はデータの中から必要なビットを取り出すために使います。

たとえば、データの最上位ビットが1でなく、1にしたいときは次のようにします。(図8)

ついでに最上位ビットを0にするとき、上位4ビットを反転させ、下位4ビットをそのままにするときをそれぞれ図7にまとめます。

## ●実際の各命令

ロード命令と同様に8080の命令と比較して説明したいと思います。まず8ビット関係の演算のOPコードの表を見てください。白地の部分が8080で使われていた命令です。灰色の部分がZ-80で増えた部分です。

新命令は、インデックス・レジスタを使った命令です。ロード命令の所でも説明したと思いますが、レジスタAの相手になるデータには、インデックス・レジスタの内容に命令コードでdと書かれている符号つき8ビット2進数を加えたアドレスを持ったメモリの1バイトが使われます。(表1)

## ●16ビット演算

表2を見てください。色分けは8ビット演算と同じ



表 6

| LOC | INS | LABEL | O P  | OPERAND  | COMMENT |
|-----|-----|-------|------|----------|---------|
| 0   |     |       | LD   | HL, 2000 |         |
| 1   |     |       | LD   | B, HL    |         |
| 2   |     |       | INC  | HL       |         |
| 3   |     |       | LD   | C, HL    |         |
| 4   |     |       | INC  | HL       |         |
| 5   |     |       | LD   | HL       |         |
| 6   |     |       | INC  | HL       |         |
| 7   |     |       | LD   | E, HL    |         |
| 8   |     |       | EX   | DE, HL   |         |
| 9   |     |       | ADD  | HL, BC   |         |
| A   |     |       | EX   | DE, HL   |         |
| B   |     |       | INC  | HL       |         |
| C   |     |       | LD   | (HL), B  |         |
| D   |     |       | INC  | HL       |         |
| E   |     |       | LD   | (HL), E  |         |
| F   |     |       | HALT |          |         |

表 5

| ADD命令        | 計算式            | 注             |
|--------------|----------------|---------------|
| ADDA, B      | B ← B + A      | A + B の和      |
| ADDA, 17H    | A ← B + 17H    | A + 17H の和    |
| ADDA, HL     | A ← A + HL     | A + HL の和     |
| ADDA, X + 2H | A ← A + X + 2H | A + X + 2H の和 |

表 7

| LOC | INS | LABEL | O P  | OPERAND  | COMMENT |
|-----|-----|-------|------|----------|---------|
| 0   |     |       | LD   | HL, 2001 |         |
| 1   |     |       | LD   | A, (HL)  |         |
| 2   |     |       | INC  | HL       |         |
| 3   |     |       | INC  | HL       |         |
| 4   |     |       | ADD  | A, HL    |         |
| 5   |     |       | INC  | HL       |         |
| 6   |     |       | INC  | HL       |         |
| 7   |     |       | LD   | (HL), A  |         |
| 8   |     |       | DCR  | HL       |         |
| 9   |     |       | DCR  | HL       |         |
| A   |     |       | DCR  | HL       |         |
| B   |     |       | LD   | A, (HL)  |         |
| C   |     |       | DCR  | HL       |         |
| D   |     |       | DCR  | HL       |         |
| E   |     |       | ADC  | A, (HL)  |         |
| F   |     |       | HALT |          |         |

ます。答を2004H～2005Hに入れるとします。それぞれの数は32768より小さい数とします。まず、16ビットレジスタを使ったプログラムを作りましょう。(表6) また、8ビットのレジスタを使うとこのようになります。(表7)

## 参考文献

- 1) トランジスタ技術1977年、9月号、Z-80の徹底研究 パート2
- 2) Z-80マニュアル



さて、御立ち合い。たった3行のBAS I Cプログラムで、TVのド真中にデジタル時計を表示してみせましょう。

プログラムの各行について説明すると、まず全プログラムを入力したあと、RUNをかけると、第1行目でH(時)、M(分)、S(秒)を聞いてくるので、それぞれ現在の時刻をキーインしてやります。TK-80BSにはカーソル機能があるので、これをフルに活用して、満字の時、分、秒を画面所定の位置に出させます。

第2行目は、60進と24進で、S、M、HをIF (GOTO) 文を使って加算、桁上げさせるループを作ります。第3行目のFOR-NEXT 文は単純なタイマーとして機能し、1ループを正確に1秒の周期をかけて、まわるような時限をとらせます。実際には、個々のTK-80のクロック周波数の多少の違いや、または

## TK-80BS用 デジタル時計 プログラム

行番号を増やしたりしてプログラムを組み替えるようなときは、このタイマーの1092を変更して、時計の進み、遅れを調整します。

本プログラムの応用として、AM、PMの12時間表示、日付曜日表示も簡単にできますが、プログラムが長くなると、実行ループの時間がかかりすぎて、時計表示が調整しきれなくなり、実用にならなくなってしまう。

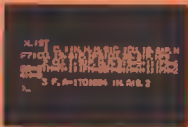
I/O 2月号のカレンダー・プログラムに、このデジタル時計プログラムを変更追加して、まともな、お洒落なTV表示になります。(脚間黒 今P太)



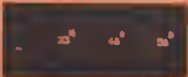
自作のケースに納めたTK-80BS



プログラム・リスト



プログラム実行例



## 送信

手持ちのマイコンを「LED点灯装置」にしないために

8080  
モールス符号の発生

手持ちのマイコンを「LED点灯装置」に終わらせないためにモールス符号発生器を作ってみました。

原回路<sup>1)</sup>はタッチ・コードとして、タイプで入力した文字を、モールス符号にして送出するものですが、この装置はそれをマイコンにさせようというものです。

動作原理の詳細は上の本を見てもらうとして、その概要を簡単に述べます(図1)。

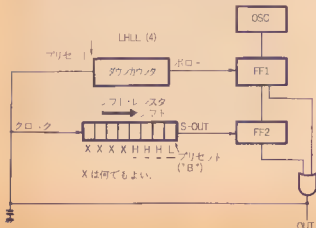
モールス符号の長点を「L」に、短点を「H」に対応させ、シフトレジスタ(SR)にプリセットします。一方、長点数と短点数の合計をカウンタにプリセットします。図1はB(1011)の場合を示しています。SRの出力はFF2のリセットを制御し、長点短点をそのつど切り換えます。カウンタはダウン動作をしており、長短点を、プリセットした数だけ送るとゼロを出してFF1を止め、全体がストップします。なお、SRおよび、カウンタの動作クロックは符号出力から取り出しています。

このように1つの符号に対して2バイト使用しています。この対応表の1例を表1に示します。

## 回路

本回路に使用したアドレスおよびその用途は表2に示します。回路図およびタイムチャートを図2、図3に示します。

図1 符号送出の原理



藤 英一

BUSY FLAGはアナログSWからデータ・バスにつないでいますが、3ステートのバッファを使用した方がよいと思います。

本器はカウンタに、C-MOSでよいものがなかったため、TTLを用い、その入力にバッファとしてCMOSインバータを付けています。このためにデータが反転するので、プログラムを作る時には注意が必要です。OUTキャンセル・ヒットは、FF1、FF2を動作させたまま、出力だけ出さないようにするもので、連続して符号を送出する際の符号と符号の間隔(3短点)とか、スペースのために設けてあります。

図中のCは、長点送出の際発生するヒゲを除くものですから、必ず入れて下さい。これがないと確実に誤動作をおこします。

発振回路の定数は、かなり速い符号を送出します。というのは、定数を変えて速度を落した場合、符号の感じが自然でなくなるからです。遅い速度で動作させる場合にはウェイト・コントロールを付けて、長短点の比を変えてやる必要があります。

## 用途

本器の出力は低周波発振器(モールス練習用)を制御していますが、バッファを付けて、送信機をコントロールすることも可能です。現在は写真のように単なる音出しのみですが、フルキーボード、CRTなどのある人は、さらに高度な使用方法が考えられると思います。

最後にプログラムの一例と、そのフローチャートを、表3、図4に示します。これは特定のエリアに書き込んだデータとして送出するもので、END記号(00)を検出して再度データの送出をくり返します。

## 参考文献

1)「ROMを使ったエレキー」, CQ, 1977. 10月号, CQ出版





TTLレベルでコントロールできる低周波発振器の例

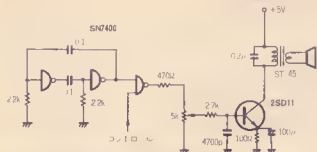
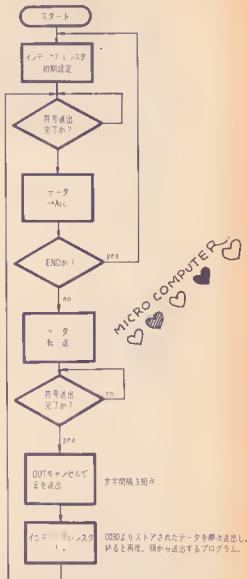


表3 プログラムの一例

| ス    |       |        |       |                   |              |
|------|-------|--------|-------|-------------------|--------------|
| 0000 | START | LXI H  | 0030  | インテックス・レジスタ初期設定   |              |
| 0003 | FLAG1 | IN     | FF    |                   |              |
| 0005 |       | ORA A  |       |                   |              |
| 0006 |       | JPE    | FLAG1 |                   | BUSYフラグ・チェック |
| 0009 |       | MOV AM |       |                   |              |
| 000A |       | SUI    | 00    |                   | END記号チェック    |
| 000C |       | JZ     | RPT   |                   |              |
| 000F |       | OUT    | FF    |                   |              |
| 0011 |       | INX H  |       |                   |              |
| 0012 |       | MOV AM |       |                   | データ送込        |
| 0013 |       | OUT    | FE    |                   |              |
| 0015 | FLAG2 | IN     | FF    |                   |              |
| 0017 |       | ORA A  |       |                   |              |
| 0018 |       | JPE    | FLAG2 |                   | BUSYフラグ・チェック |
| 001B |       | MVI A  | FE    |                   |              |
| 001D |       | OUT    | FF    |                   | 文字と文字の間隔を3桁に |
| 001E |       | MVI A  | EE    |                   | あける。         |
| 001F |       | OUT    | FE    |                   |              |
| 0021 |       | INX H  |       |                   |              |
| 0024 |       | JMP    | FLAG1 |                   |              |
| 0027 | RPT   | JMP    | START |                   |              |
| 130  |       | DC1    |       | 出力データ20030より書き込みず |              |
|      |       | DC1    |       | ( 終りの符号は00である     |              |
|      |       | DC2    |       |                   |              |
|      |       | DC2    |       | データーア             |              |
|      |       | DC3    |       |                   |              |
|      |       | DC3    |       |                   |              |
|      |       | DC4    |       |                   |              |
|      |       | DC4    |       |                   |              |
|      |       | DC5    |       |                   |              |
|      |       | DC5    |       |                   |              |
|      |       | DC6    |       |                   |              |
|      |       | DC6    |       |                   |              |
|      |       | DC7    |       |                   |              |
|      |       | DC7    |       |                   |              |
|      |       | DC8    |       |                   |              |
|      |       | DC8    |       |                   |              |
|      |       | DC9    |       |                   |              |
|      |       | DC9    |       |                   |              |
|      |       | DC10   |       |                   |              |
|      |       | DC10   |       |                   |              |
|      |       | DC11   |       |                   |              |
|      |       | DC11   |       |                   |              |
|      |       | DC12   |       |                   |              |
|      |       | DC12   |       |                   |              |
|      |       | DC13   |       |                   |              |
|      |       | DC13   |       |                   |              |
|      |       | DC14   |       |                   |              |
|      |       | DC14   |       |                   |              |
|      |       | DC15   |       |                   |              |
|      |       | DC15   |       |                   |              |
|      |       | DC16   |       |                   |              |
|      |       | DC16   |       |                   |              |
|      |       | DC17   |       |                   |              |
|      |       | DC17   |       |                   |              |
|      |       | DC18   |       |                   |              |
|      |       | DC18   |       |                   |              |
|      |       | DC19   |       |                   |              |
|      |       | DC19   |       |                   |              |
|      |       | DC20   |       |                   |              |
|      |       | DC20   |       |                   |              |
|      |       | DC21   |       |                   |              |
|      |       | DC21   |       |                   |              |
|      |       | DC22   |       |                   |              |
|      |       | DC22   |       |                   |              |
|      |       | DC23   |       |                   |              |
|      |       | DC23   |       |                   |              |
|      |       | DC24   |       |                   |              |
|      |       | DC24   |       |                   |              |
|      |       | DC25   |       |                   |              |
|      |       | DC25   |       |                   |              |
|      |       | DC26   |       |                   |              |
|      |       | DC26   |       |                   |              |
|      |       | DC27   |       |                   |              |
|      |       | DC27   |       |                   |              |
|      |       | DC28   |       |                   |              |
|      |       | DC28   |       |                   |              |
|      |       | DC29   |       |                   |              |
|      |       | DC29   |       |                   |              |
|      |       | DC30   |       |                   |              |
|      |       | DC30   |       |                   |              |
|      |       | DC31   |       |                   |              |
|      |       | DC31   |       |                   |              |
|      |       | DC32   |       |                   |              |
|      |       | DC32   |       |                   |              |
|      |       | DC33   |       |                   |              |
|      |       | DC33   |       |                   |              |
|      |       | DC34   |       |                   |              |
|      |       | DC34   |       |                   |              |
|      |       | DC35   |       |                   |              |
|      |       | DC35   |       |                   |              |
|      |       | DC36   |       |                   |              |
|      |       | DC36   |       |                   |              |
|      |       | DC37   |       |                   |              |
|      |       | DC37   |       |                   |              |
|      |       | DC38   |       |                   |              |
|      |       | DC38   |       |                   |              |
|      |       | DC39   |       |                   |              |
|      |       | DC39   |       |                   |              |
|      |       | DC40   |       |                   |              |
|      |       | DC40   |       |                   |              |
|      |       | DC41   |       |                   |              |
|      |       | DC41   |       |                   |              |
|      |       | DC42   |       |                   |              |
|      |       | DC42   |       |                   |              |
|      |       | DC43   |       |                   |              |
|      |       | DC43   |       |                   |              |
|      |       | DC44   |       |                   |              |
|      |       | DC44   |       |                   |              |
|      |       | DC45   |       |                   |              |
|      |       | DC45   |       |                   |              |
|      |       | DC46   |       |                   |              |
|      |       | DC46   |       |                   |              |
|      |       | DC47   |       |                   | </           |

図4 フローチャート



de BUG

●I/O '78, 5月号“LEXII-16 用差アセンブラ”の中に訂正があります。

- 1) P77 表1 EOR 60 は E0 の誤り  
2) P77 ⑥ M・三三 は M→三の誤り  
3) P77 ① 1, 1, 6, 1 は 1, 9, 6, 1  
4) P77 ⑥ 0 → 7 7 7 7 7 7 7 7 は 7 7 7 7 7 7 7 7 の誤り

- 5) P77 ④ [A] — [DISP] の [A] は ADD 命令の A
- 6) P80 1 行目 X0 は X0 の誤り  
3 行目 X0 は X1 の誤り
- 7) P81 フローチャート 3 は、フローチャート 2 のつづきである。
- 8) P83 處アセンブラリスト 1 行目 コメント欄  
“プログラム・スタート・アドレス”は削除
- 9) P83 アドレス 1214 のコメント欄に  
“RN1 の引数”を追加

●T N-80 B S 最新情報③

- P118 下から3行目「3AF0」は「1AF0」の誤り

## 受信

H68/TRを用いた  
モールス符号の解読

大西 義純

モールス符号によるマン・マシン・インターフェイスは、ある程度実用的で最もシンプル（入力には1個のスイッチ(電鍵のこと)、出力にスピーカを用いる)なので、単にモールス符号を解読する以上の用途が考えられると思います。

マイコンの出力にはCRTを用いるとしても、電鍵をたたいて「トン・ツー」で入力するのはいかがでしょうか。

手始めにH68/TRを用いてモールス符号を解読する実験を行ないましたので、ここにその全容を報告します。

## 仕様

できる限り簡単にするために次のように決めました。

①コンソールのディスプレイを用いて独得のアルファベットで表示を行なう。従って、解読する符号は欧文のみとなる。

②モールス符号の入力には電鍵を用いる。

(ミニチュア・リレーなどを用いることにより、オーディオ信号を入力とすることもできる。)

③付加するハードは基本的には単純なものとして解読のすべてをソフト行なうようにしたが、実験が容易に行なえるように低周波発振回路(モニタ)、アクセス表示回路を追加した。

## 構成

本実験に用いた装置は次のとおりです。

- ①H68/TR本体：RAMは基本容量(1Kバイト)で充分です。(プログラム、データ・テーブルなどを合わせて256バイトに納まりました。)バスマイバが必要。
- ②コンソール：表示に使用
- ③電源：付加回路の電源はH68/TRのバスから取っている。
- ④電鍵：モールスの練習用のもの
- ⑤スピーカ：モニタとして使用
- ⑥付加回路：次項に詳細を示す。

## 付加回路

図1に全回路を示します。

標準のTTLで構成し、①アドレス・デコーダ②データ回路③アクセス表示回路④低周波発振回路から構成されています。

## ①アドレス・デコーダ

アドレスは\$8000番地に割当てましたが、7400の入力に付けるインバータが少なく済むために他に理由はありません。

なお、H68/TRのハードウェア・マニュアルp.46のバス配列表では24番ピンA面はVMAとなっていますが、これは間違いで前ページの回路図通りVMAが正しいようです。

## ②データ回路

キー入力にはR、Cでチャタリングを除去した後、シュミットトリガ7413で波形整形してモニタ回路のゲートとラッチ回路に与えられます。

出力にはオープン・コレクタの7401を用いました。データは1ビットだけで、D<sub>0</sub>のみを用いています。

## ③アクセス表示回路

DタイプF/F 7474を用いて\$8000番地がアクセスされるたびに2つのLEDが交互に点滅するようにしました。7474の入力の330Ωの抵抗と、330pFのコンデンサはデコード成分に含まれているヒゲを除去するた

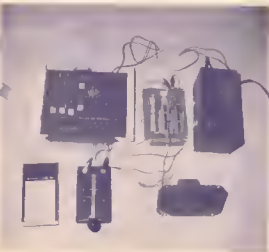
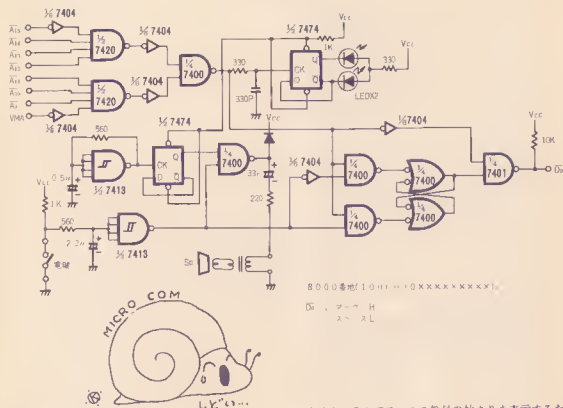


圖 1 同踏圖



めのものです。

この回路は機能に関係するものではありませんが、回路の試験などを行なう時に役立ちました。

#### ④ 低周波発振回路

モールス符号をモニタするために設けました。シュミットトリガ7413を用いて発振させ、7474でデューティ比を $\frac{1}{2}$ にしています(周波数も $\frac{1}{2}$ になります)。

7490のゲートを通してキーが押されている時にスピーカから低周波音が出ます。

## ソフトウェア

フローチャートを図2に、プログラム・リストを表1に示します。

プログラムの細かい説明は省いてポイントだけを以  
に述べます。

### ① メールス符号データの構成

1 文字分のモールス符号を1 バイトのデータに変換  
します。

-(トン)を1, -(ツ-)を0とするとA(ー)は10となり、モールス符号データは残りのビットすなわちD<sub>7</sub>~D<sub>2</sub>を0とすると\$02となります。

しかし、これではK(一-)も\$02となり区別が付き

ません。そこでモールス符号の始まりを表示するために符号の直前のビットを1とします。このようにすればAとKはそれぞれ\$06, \$0Aとなり区別が付きま

このようにして得られた符号データを\$02BF番地に格納しています。

### ②モールス符号の長さの判別

符号の長さを判別するための時間の基準として WA  
1 命令でタイマー割り込みを間接的に用いています。

一連の処理が終了すればWA1命令で待機させておき、タイマ割り込みが生じた回数をカウントして、所定の回数（ここでは12回すなわち10ms）になれば\$8000番地のデータD<sub>0</sub>を読み出しています。

読み出したデータが前のデータと同じであれば\$02 BD番地の内容をインクリメントします。前のデータと異なる時は所定のルーチンへジャンプします。

なお 802B D 番地の D<sub>7</sub> ビットはマークかスペースかの区別に用いています。

モールス符号の長さの基準は表2のとおりです。

### ④ 解法お上り表示

\$02BF番地のモールス符号データをインテックス修飾によって、直接セグメント・データに変換し、コンソールのディスプレイで表示させています。

表示の方法は電光ニュースのように文字が左から右



表1 プログラムリスト

| アドレス | コメント    | 操作  | メモリー        | コメント                  |
|------|---------|-----|-------------|-----------------------|
| 0200 | CE 02BD |     | LDA #502BD  | プログラム開始番地             |
| 03   | 6F 00   |     | CLR 0, X    |                       |
| 05   | 6F 02   |     | CLR 2, X    |                       |
| 07   | 86 05   |     | LDA #5      | イニシャライズ               |
| 09   | B7 E007 |     | STAA \$E007 |                       |
| 0C   | 0E      |     | CLI         |                       |
| 0D   | 4F      | L04 | CLRA        |                       |
| 0E   | 3E      | L09 | WAI         |                       |
| 0F   | 4C      |     | INCA        | タイマールーチン              |
| 10   | 81 0C   |     | CMPA #50C   | 10ms                  |
| 12   | 26 FA   |     | BNE L09     |                       |
| 14   | A6 00   |     | LDA 0, X    |                       |
| 16   | 2A 10   |     | BPL L01     | 符号長メモリの内容がスペースのときノランチ |
| 18   | F6 8000 |     | LDA \$8000  |                       |
| 1B   | 27 20   |     | BEQ L20     | モールス信号かマクのときランチ       |
| 1D   | 81 92   |     | CMPA #592   | マークの長さが10ms×512をこえる   |
| 1F   | 22 04   |     | BH1 L03     | とき(ワーのとき)ランチ          |
| 21   | C6 01   |     | LDA #1      |                       |
| 23   | 20 21   |     | BRA L21     | トップ記号を\$02BF番地に入れる    |
| 25   | 5F      | L03 | CLRB        |                       |
| 26   | 20 1E   |     | BRA L21     | トップ記号を\$02BF番地に入れる。   |
| 28   | F6 8000 | L01 | LDA \$8000  |                       |
| 28   | 26 10   |     | BNE L20     | モールス信号がスペースのときランチ     |
| 2D   | 81 12   |     | CMPA #512   | スペースの長さが10ms×512以下    |
| 2F   | 23 1D   |     | RLS L22     | のとき(符号間のスペースのとき)ランチ   |
| 31   | 8D 26   |     | BSR L08     | 解読表示サブルーチンへランチ        |
| 33   | A6 00   |     | LDA 0, X    |                       |
| 35   | 81 39   |     | CMPA #539   | スペースの長さ10ms×539以下     |
| 37   | 23 15   |     | RLS L22     | のとき(文字間のスペースのとき)ランチ   |
| 39   | 8D 1E   |     | BSR L08     |                       |
| 3B   | 20 11   |     | BRA L22     | 語間の空白を表示              |
| 3D   | 48      | L20 | ASLA        |                       |
| 3E   | 81 FE   |     | CMPA #5FE   |                       |
| 40   | 27 02   |     | BEQ L26     | 符号長メモリ・インクリメントルーチン    |
| 42   | 6C 00   |     | (INC 0, X   | (\$02BD)              |
| 44   | 20 C7   | L26 | BRA L04     |                       |
| 46   | A6 02   | L21 | LDA 2, X    |                       |
| 48   | 26 04   |     | BMI L22     |                       |
| 4A   | 48      |     | ASLA        | モールス符号挿入ルーチン          |
| 024B | 1B      |     | ABA         |                       |
| 4C   | A7 02   |     | STAA 2, X   |                       |
| 4E   | A6 00   | L22 | LDA 0, X    |                       |
| 50   | 88 80   |     | EORA #580   | 符号長メモリ・イニシャライズ        |
| 52   | 84 80   |     | ANDA #580   | ルーチン                  |
| 54   | 4C      |     | INCA        | (Dnビットを反転し De-Dn      |
| 55   | A7 00   |     | STAA 0, X   | ビットの内容を\$01とする)       |
| 57   | 20 B4   |     | BRA L04     |                       |
| 59   | A6 02   | L08 | LDA 2, X    | 解読表示サブルーチン(=502A1)    |
| 5B   | 2B 1F   |     | BMI L08     | Dnビットが(エラー)のときランチ     |
| 5D   | 85 E0   |     | BITA #5E0   | 6ビット以上のデータのときランチ      |
| 5F   | 26 1F   |     | BNE L0C     |                       |
| 61   | FE 02BE | L0F | LDA \$02BE  | (02BE)=02(02BF)=符号データ |
| 64   | A6 C0   |     | LDA \$C0, X | セグメントデータをロード          |
| 66   | CE E821 | L0D | LDA #E821   |                       |
| 69   | E6 00   | L25 | LDA 0, X    |                       |
| 6B   | A7 00   |     | STAA 0, X   | 表示ルーチン                |
| 6D   | 17      |     | TBA         |                       |
| 6E   | 09      |     | DEX         |                       |
| 6F   | 8C E813 |     | CPX #5E813  |                       |
| 72   | 26 F5   |     | BNE L25     |                       |
| 74   | CE 02BD | L15 | LDA #502BD  |                       |
| 77   | 86 01   |     | LDA #1      | 符号データ・メモリ(\$02BF番地)   |
| 79   | A7 02   |     | STAA 2, X   | イニシャライズ、メインルーチンへ復帰    |
| 7B   | 39      |     | RTS         |                       |

| アドレス | ヘキサ     | 命令  | コメント         |                                             |
|------|---------|-----|--------------|---------------------------------------------|
| 027C | 86 84   | LDB | LDA #84      | e(エラー)をロード → フランチ                           |
| 7E   | 20 E6   |     | BRA LDB      |                                             |
| 80   | 81 6A   | LDC | CMPA #56A    | 6Aでないときフランチ                                 |
| 82   | 26 0A   |     | BNE LOE      |                                             |
| 84   | 86 E821 |     | LDA #E821    | ピリオドを表示してフランチ                               |
| 87   | 84 7F   |     | ANDA #57F    |                                             |
| 89   | 87 E821 |     | STA #E821    |                                             |
| 8C   | 20 E6   |     | BRA L15      |                                             |
| 8E   | CE 02DF | LDE | LDX #02DF    | アドレス交換ルーチ                                   |
| 91   | C6 1F   |     | LDA #51F     |                                             |
| 93   | 08      | L18 | INX          | 概当するコ → がないときフランチ                           |
| 94   | 5C      |     | INCB         |                                             |
| 95   | C1 30   |     | CMFB #30     | 発しないときフランチ                                  |
| 97   | 27 E3   |     | BEQ LDB      |                                             |
| 99   | A1 10   |     | CMPA \$10, X | フランチしてAとBの内容(交換されたアドレス)<br>をXレジスタの2バイト目にSET |
| 9B   | 26 F6   |     | BNE L18      |                                             |
| 9D   | F7 02BF |     | STAB \$02BF  |                                             |
| A0   | 20 BF   |     | BRA LDF      |                                             |

02A2~02BC : ブランク

ワーキングレジスタ

02BD : 符号長データ格納

02BE : 02

02BF : モールス符号データ格納

データテーブル

| アド   | シンボル | 文字/符号データ | アドレス | シンボル | 文字/符号データ | アドレス | シンボル | アドレス | シン | キ   |
|------|------|----------|------|------|----------|------|------|------|----|-----|
| 02C0 | FF   | F / 00   | 02D0 | 84   | e / 10   | 02E0 | C0   | 02F0 | 20 | 0   |
| C1   | FF   | フ / 01   | 01   | 84   | e / 11   | E1   | 90   | F1   | 21 | 9   |
| C2   | CE   | T / 02   | D2   | 98   | Q / 12   | E2   | 80   | F2   | 23 | 8   |
| C3   | 86   | E / 03   | D3   | 86   | Z / 13   | E3   | D8   | F3   | 27 | 7   |
| C4   | AA   | M / 04   | D4   | 91   | Y / 14   | E4   | 82   | F4   | 2F | 6   |
| C5   | AB   | N / 05   | D5   | C6   | C / 15   | E5   | F9   | F5   | 30 | 1   |
| C6   | 88   | A / 06   | D6   | 95   | X / 16   | E6   | A4   | F6   | 38 | 2   |
| C7   | E6   | I / 07   | D7   | 83   | S / 17   | E7   | 80   | F7   | 3C | 3   |
| C8   | A2   | O / 08   | D8   | E1   | J / 18   | E8   | 99   | F8   | 3E | 4   |
| C9   | C2   | G / 09   | D9   | 8C   | P / 19   | E9   | 92   | F9   | 3F | 5   |
| CA   | 85   | K / 0A   | DA   | 84   | e / 1A   | EA   | F6   | FA   | 47 | :   |
| CB   | A1   | D / 0B   | DB   | C7   | L / 1B   | EB   | F3   | FB   | 4C | :   |
| CC   | E2   | W / 0C   | DC   | 84   | e / 1C   | EC   | F0   | FC   | 52 | ( ) |
| CD   | 8F   | R / 0D   | DD   | 8E   | F / 1D   | ED   | AD   | FD   | 2D | :   |
| CE   | C1   | U / 0E   | DE   | 81   | V / 1E   | EE   | AC   | FE   | 73 | ?   |
| CF   | 93   | S / 0F   | DF   | 89   | H / 1F   | EF   | 00   | FF   | 00 | 予備  |

セグメントコード

セグメントコード

符号データ  
対応しています。

させてモールス信号を入力させた結果、ほぼ満足に解読することができました。

本プログラムでは次の2つの問題点がありますので、それを示してこのリポートを終わります。

① WAI 命令を用いているので前のプログラムの処理が長びいた場合割り込みが生じてタイムラウチンのインクリメントが飛ばされる可能性があります。割り込みの周期は0.8msなので、ほとんど問題にならないと思います。

② フローチャートを見ればわかるように、次のマーク信号が来て初めてそれまでのスペースの長さを判別し、前の文字を表示するようになっています。したがって最後の文字は表示されません。少し気になる点です。

表2 モールス符号の長さ

| 符号の種類 | 短音を1とした時の比 |
|-------|------------|
| マーク   | 1          |
| スペース  | 3          |
| 符号間   | 1          |
| 文字間   | 3          |
| 単語間   | 7          |

お  
決定!

## 基礎力養成講座

## 電子回路入門

マイコンファンならぜひマスターしたい!

## 第3回

## 電源

について  
考える

江口 敏彦

前回は一般的なことについて述べましたが、範囲が広くてピンと来なかった方が多いと思います。今回は実際に装置の源である、電源を例に取って考えて見ましょう。

通常電子回路の電源は、AC100Vを直接使用することは少なく、AC100Vをある電圧に降圧または昇圧整流して使うのが普通です。そこでデジタルIC用として5Vの定電圧電源を作ります。要求する性能はICの規格から5V $\pm$ 0.25Vで5mAを目標とします。

定電圧の方式には、出力を連続制御する方法とオン、オフを制御して所定の出力を得る断続（スイッチング）制御方式があります。この2つの方式にも一長一短（表1）があり、さらに連続制御方式は、負荷に定電圧回路が直列に入る直列制御方式（図1）と並列に入る並列制御方式（図2）に分けられますが、今回は一般的な直列制御方式で回路は図3に示します。

## 定電圧回路の効率

定電圧回路の動作は、常に入力電圧と出力電圧の差（入出力電位差）が要求されます。そして最低でも3Vは必要です（図4）。さらにAC100Vの変動マージンを見込むと、入出力電位差はもっと必要になります。直列制御方式の定電圧回路では、この入出力電位差と回路動作に必要な電力が損失になるのです。それでは、今回制作する電源を例にとって損失を調べて見ましょう。

表1 連続制御方式とスイッチング制御方式の比較

| 項目      | 方式 | 連続制御    | スイッチング制御 |
|---------|----|---------|----------|
| 効 率     |    | 40~50%  | 70~80%   |
| 安 定 度   |    | 優 良     | 良        |
| 雑 音     |    | 優 良     | 良        |
| 過 度 応 答 |    | 優 良     | 良        |
| 重 量     |    | 重 い     | 軽 い      |
| 体 積     |    | 大       | 小        |
| 価 格     |    | 5A以下は安い | 5A以上は安い  |



まず、図4の斜線部分が損失になるわけで、図5(a)の場合、入力電力が約45W、出力電力が25Wですから残り約20Wが損失になります。この入力に対する出力の割合を効率といいますが、この場合約55%です。一方、(b)の場合では約88%になります。このように同じ出力電力を得る場合にも条件によって大幅に変化します。実際はトランス、整流器その他の損失がありますので、効率は低下します。

参考に今回製作した電源と市販されているスイッチング方式の定電圧電源と比較した結果を表2に示します。外観は写真1です。

## 熱対策

表2の通り、5Aの出力を得ると、トランス、ダイオードなどの損失があるため効率が50%弱になってしまい、ほぼ小型の半田ゴテと同じ、約25Wの熱が損失として発生します。熱発生も多いものから順に並べると、トランジスタ、トランス、整流用ダイオード、IC、抵抗R<sub>i</sub>になります。この中で放熱器の必要なのは、トランジスタ、整流用ダイオード、ICです。

私の場合は、トランジスタの放熱器が小さい（1.5℃/W位で、寸法は幅100、高さ75、厚さ20mm平直使用）ので、5A負荷で長時間使うと室温より25℃~30℃上昇します。夏でもトランジスタの規格は越えませんが、相当な温度になると思われる。整流用ダイオードは3mm厚のフレームの挿入を兼ねたアルミLアンクルに取り付けます。

また、ICとSCRも別のアンクルに取り付けます。ただし、SCRは通常では電力消費はありませんが、過電圧が生じると瞬間的に大電流が流れますので、やはりアンクルに取付けます。

図1 直列制御方式の定電圧回路 図2 並列制御方式の定電圧回路





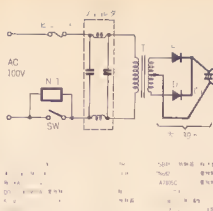
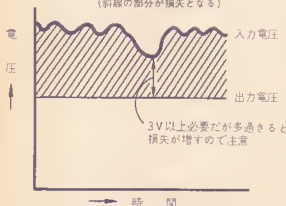


図4 定電圧回路の損失  
(斜線の部分が損失となる)



箱の出る話でアツクりましたが、電解コンデンサC1、C2など寿命の短い部品(程度の差はありますが、一般的に約10年経過すると容量の減少トラブルが発生する場合があります。私の職場では信頼度を特に要求されているので6年で交換しています。)なので、熱によってさらに短くならないように配置することです。

蛇足ですが、ジャンク品を使うときは容量が減少してないか十分に確認が必要です。時々広告や店頭で、私の会社の払い下げ品と称して売っているのを見ますので、I/Oの読者だけにソートお知らせします。ゾーと悪くなりましたか？

## 雑音

定電圧回路では出力側にハム雑音が出ても、これを電圧変動として制御するため、出力にはハム雑音が非常に少なくなります。しかし、1Cのアース線にトランスやAC100Vの誘導を受けると、話は少々変わってきます。誘導電圧は非常に少なくても、1Cで70~80dBの増幅器(約1万倍)によって、出力電圧が変動したのと同様に出力電圧を制御するので、逆に出力電圧へ大きな変動が与えられます。(配線の項参照)

そのほか、気をつけなければいけない雑音は、ホワイト雑音によるものです。ホワイト雑音(アンプなどで入力信号を与えないでVRを上げたときに出る雑音)は実用上、完全に無視できますが、取り除くことは雑音の出ないアンプを作ることと同様に困難なことです。もし問題があれば、負荷側の回路で対策すべきと考えます。今回製作の定電圧電源の雑音は、5Hz~1MHz帯域のレベル・メーターによる測定で、雑音の

図3

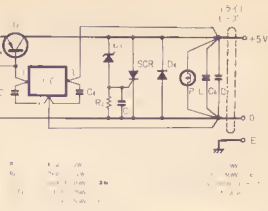


表2 スイッチング方式との効率の違い

| 負荷電流   | 1A  | 2A  | 3A  | 4A  | 5A  | 6A  | 7A  |
|--------|-----|-----|-----|-----|-----|-----|-----|
| 効率 (%) | 23% | 34% | 40% | 44% | 46% | 50% | 50% |
| 損失 (%) | 77% | 66% | 60% | 56% | 54% | 50% | 50% |

- 1. スイッチング方式はLSA型、最大7A出力時
- 効率80%以上は、2. 製作時の回路で100V、50Hz、50°C、50%湿度時

実効値は、無負荷で0.1mV、5A負荷で1.5mVです。

外部の雑音対策は、ACラインにライン・フィルタを、出力側にフェライト・ビーズとC7を入れます。多くの場合、不要と思いますが特殊な環境(電気溶接機や公称500W出力の短波送信機と同居)で使うことを考慮しました。

話はそれますが私のマイコンは前記の環境でも完全に動作します。「備えあれば憂い無し」と言うところです。どこかのコンピュータのように、古い蛍光灯が点滅しただけでデウソしては大変です。本体から外部に線が出ていると、雑音受信アンテナに早変わりのおそれがあります。本体とI/Oとの接続は、アース、シールドなど充分考慮してください。場合によっては同軸ケーブルを使っても、放送電波が混入したケースがありますので...

ところで、アメリカ製の一部のマイコンにケースがプラスチックで作られたものがありますが、雑音環境の悪いところで機械制御用に使用した場合、誤動作がないのでしょうか。逆にA・M・F・Mレシーバなどに雑音を与えないの心配です。(両方ともアルミ箔や導電塗料で対策も簡単に行なえますが)

写真1 スイッチング方式と直流制御方式との比較  
(直流型の上にあるのはタバコ)

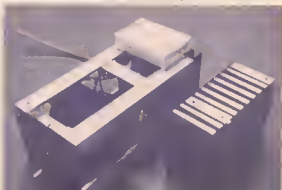


図5 出力電圧による効率の違い



図6 過渡応答の例

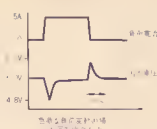
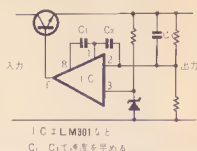


図7 応答性の良い定電圧回路例



## 過渡応答

どんな定電圧回路でも言えることですが、負荷が急に変わった場合、定電圧回路の制御は瞬時に行なわれないので負荷変動に追従できないことが起ります(図6)。応答性を良くするには、高速動作する部品、回路を使用すればある程度は向上します(図7)。

しかし、異常発振などの危険がともないますので、負荷回路が必要としているかを考えると、多くの場合は不必要と考えられます。それは定電圧回路から負荷回路までは、抵抗、容量、誘導分があるので、定電圧回路で対策するのは無意味であることが多いです。(違い定電圧回路より近くのコンデンサー??)今回、制作した電源回路は通常、使用するには充分だと思えます。

## 精度

精度は経年変化、温度変化、入力電圧変化、負荷の変化などに左右されます。一般的に、今回使用した3端子レギュレータは、ドライブ能力の低いもの、例えば、LM305などを単体で使ったものより(回路の違いや熱的結合のためか)劣ります。それでも昔、トランジスタ数個で作った定電圧回路より格段すぐれています。実測値を表3に示します。

もし、出力電圧が4.8V位るとき(定電圧ICの規格は4.8V-5.2V)は、電源配線やプリント・パターンに注意しても、TTL ICの規格(4.75V)を満足できなくなる恐れがあるので、図8のように抵抗を入れて補償します。

また、出力電流が0-5Aまで精度を良くすることは困難ですから、ゲニ抵抗を兼ねてパイロット・ランプに6.3V、

表3 図3の出力電流と出力電圧

| 出力電流 | 出力電圧    |
|------|---------|
| 0 A  | 5.059 V |
| 1 A  | 5.052 V |
| 2 A  | 5.046 V |
| 3 A  | 5.041 V |
| 4 A  | 5.037 V |
| 5 A  | 5.033 V |

図8 出力電圧を上げる場合の補償回路

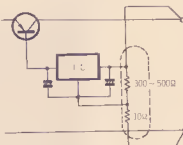
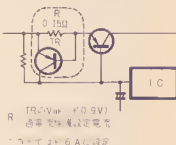
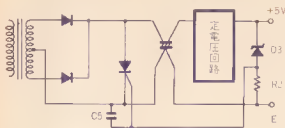


図9 電氣的な過電流保護回路



ただだめじゃ

図10 定電圧回路に無理を与えない過電圧保護回路



電圧が発生しても保護されます。なお、動作電圧は、ツェナーダイオードの電圧+0.8V~0.9Vなので約6Vです。また、SCRを入れるところを図10のようにすると多様な電源も保護でき、SCRも小型のものが使えます。費用もリレー、ダイオード（小電流用で良い）が増えても、SCRが安くなりますのでためてははいかがでしょうか。

保護回路動作の簡単な確認方法は、220V/F16W以上のコンデンサを図3のC1、C2の端子（約9V）で充電し定電圧回路の出力端子へ、+、-を間違えず接続すれば瞬間にヒューズが飛んで確認（非常に原始的）できます。

## 部品

①ヒューズ セットに組み込むならば、前記のヒューズでも良いと思いますが、実験やそのほかショートの可能性の大きい場合は、バイメタル式のブレーカが良いと思います。ただ、ヒューズより動作が遅れます。

②ノイズ・フィルタ、フェライト・ビーズ（写真2）これらは絶対に必要とは言えませんが、つまらない雑音で誤動作するより最初から対策しておいた方が得策です。

③スパーク・キラー これも②と同様、SWの接点保護と雑音防止が目的であれば、0.03μFのコンデンサと100Ω位の抵抗を直列にすればでき上ります。

④トランス 最も効率に関係する部品で、2次出力が高いと効率が悪くなり、むずかしいところです。私の場合は出力が10V×2.6Aで入力は110Vタップを使用しました。

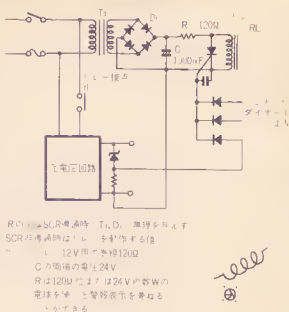
⑤整流器 300V、20Aと言うのを使用しましたが、コンデンサに大容量を使うので端子は低くてもかまいませんが、電流は充分余裕を見てください。

⑥電解コンデンサ 図3のC1、C2の容量によって同じトランスを使用しても最大出力電流が変わります。図12のように、5.5Aの出力を得る場合、10,000μFではリップルが発生するが、20,000μFではリップルは発生しないと言うことがあります。

⑦トランジスタ PNP型で電流容量が大きく、かつ大電流までh<sub>FE</sub>（直流電流増幅率）の低下がなく、V<sub>CE</sub>（コレクタ・エミッタ飽和電圧）が低いと言うことに付け加えて価格も安いこと、と言う条件から一洋電気の2SB410（ゲルマニウム・タイプ）を採用しましたが、同等の物なら何でも結構です。ただし、V<sub>CE</sub>の高い物を使うと、その分だけ出力電圧の高いトランスが要求されるので、定電圧電圧の効率が低下します。参考に図13に2SB410の特性を示します。

⑧IC おなじみ3端子レギュレータです。国内外各社より売り出されていますが、ここではフェアチャイルド社の

図11 小型SCRで多様な電源の過電圧保護例



⑨A7805Cを使用しました。A7805と比較して、入力電圧が7Vより保障されている（Cの付かないのは8Vから）ので、今回のようにトランスの出力電圧を低くおきえて効率向上をねらう場合、トランジスタとV<sub>CE</sub>とともに重要な要素になります。

⑩ツェナー・ダイオード 過電圧保護用定電圧はツェナー・ダイオードで決まります。前にも書きましたがツェナー電圧は+0.8~0.9Vになります。

⑪SCR 過電圧が発生した場合、回路を短絡するので瞬間に大電流が流れます。私の場合、ジャンク品を使用しましたが、新品を買って高価なので図11のように使えば1Aクラスでまにあいます。

⑫ダイオードD4 D4は、なんらかの原因で過電圧が発生写真2 ノイズ・フィルタを種AC100Vに右上、DC 5V出力に左下を使用



図12 コンデンサの容量変化による定電圧回路の出力変化  
(5.5A出力時)

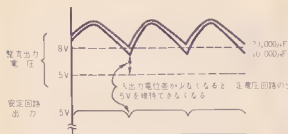
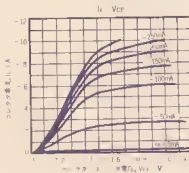
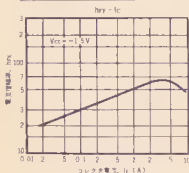


図13 2SB410の特性

絶対最大定格  $T_a = 25^\circ\text{C}$

| 項目           | 記号        | 2SB410                             | 単位               |
|--------------|-----------|------------------------------------|------------------|
| コレクタ・ベース間電圧  | $V_{CB0}$ | -135                               | V                |
| コレクタ・エミッタ間電圧 | $V_{CE0}$ | ( $I_{BE} = 500\mu\text{A}$ ) -135 | V                |
| エミッタ・ベース間電圧  | $V_{EB0}$ | -5                                 | V                |
| コレクタ電流       | $I_C$     | -15                                | A                |
| コレクタ損失       | $P_C$     | ( $T_a = 25^\circ\text{C}$ ) 40    | W                |
| 接合部温度        | $T_j$     | 85                                 | $^\circ\text{C}$ |
| 保存温度         | $T_{stg}$ | -55 ~ +85                          | $^\circ\text{C}$ |



生した場合、定電圧回路と負荷回路を保護する目的で、電流容量は1.5Aのものを使用しましたが、1A用でも良いと思います。

## 配線

配線のやり方によって、できる性能に差がでます。写真3は雑音をシンクロで拡大したところです。トラブルまでは行きませんが同じ部品で、このような簡単な回路を作っても差がでます。このほか、部品配置も重要なことですが、結果として安定度、過渡応答も差がでてくるので、配線もおろそかにはできません。

配線をする場合、おもな要点は次の通りです。

●トランス、整流器、ダイオード、コンデンサ  $C_1$ 、 $C_2$  は熱対策を考へながら、太い線できちんと配線します。そしてこのループは、できるだけ小さくし、ほかへの誘導を最小限に少なくします。これで、電圧降下を防ぎますし、インピーダンスの上昇を防ぐことができます。

●AC100Vライン、トランスの誘導に注意します。特に定電圧ICのアース線は、誘導を受けないよう距離を離します。やむを得ず近くを通るときは、図14のようにトランスのコアに直角になるように、またAC100Vとは直交させます。シールド

写真3 定電圧ICのアース線の悪い例(上)と良い例(下)

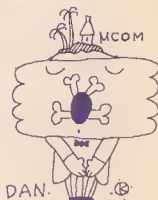
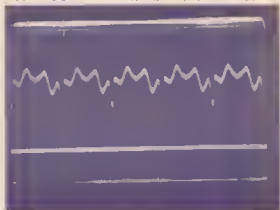


図14  
トランス誘導  
を防ぐ配線方法



できるだけトランスから離し  
コアと直角に配線する。  
シールドにも効果がない

ド線は使用しても、このような誘導に対してはほとんど効果がありません。

●図3の回路は実際の配線に近い形で書いてありますので要点を理解してください。

## 参考文献

- 1) 三洋半導体ハンドブック、誠文堂新光社
- 2) フェアチャイルド、パワーゲートブック
- 3) トランジスタ技術、1972年7月

## スイッチング・レギュレータで

電流容量 3A 以上!

マイコン用電源 +12V  
+5V  
-5V  
の製作 一條 博

あちらこちらの本を読み、自分だけのマイコンが欲しくなったとき、まず先立つものは何でしょう。まずは天下のまわりもの“お金”でしょうが、その次はというと……それが電源なのです。TK-80でもLKIT-8でも、8万なにがしの金を使っても、これがなければ絶対にうごきません。どんなにソフトに凝ろうと、かつこよくハードを作ろうと、これがなければ……という電源の製作例を紹介しましょう。

## スペック

まず、どの位の大きさで、どのような機能のものにするかをきましましょう。

当然+5V電圧は、絶対に必要です。容量は、やはりBASICが動かせる位のシステムに電力を供給できる位のものが欲しいので、最低でも3A以上とします。これ以外の電圧としては、最近の+5V単のPROM2758などですべてハードを作りあげればよいのですが、あまりにも使える部品に制限が多いことや、8080用に使えなくなるため、+12V、-5Vの電圧も用意します。これによりD-RAMも、2708のように広く使われているPROM（当然、2758よりはるかに安価）も使え、トータルではコスト安になると考えられます。

+12V、-5Vの電流容量は、容易に入手できる3端子レギュレータが1A<sub>max</sub>であるため、自動的に1Aまでとしました。この程度あれば、D-RAM16K+ROM2K程度のシステムに対して充分でしょう。スペックをまとめると次のようになりました。

+5V……3A以上  
+12V……1A  
-5V……1A

## 回路方式

+12V、-5Vの部分については、+5VからDC-DCコンバータで作ることも考えましたが、部品の入

手が間に合わなかったのと、読者の皆さんがコピーして作った場合に、部品が入手しにくい（本当は面倒なためで……近々、-5V側で0.1A位のDC-DCの例を発表しましょう。）、3端子レギュレータで済ませています。

+5Vですが、少なくとも3Aということになると、3端子型ではかなり高価かつ放熱に苦しむことになります。

トランスの巻線の都合で+12Vと+5Vを同一巻線より得ることになると、ますますレギュレータに対する負担が重くなります。いすれにせよ、シリーズ型ではディスクリットで作っても大きくなり、発熱も大きく色々な意味で不経済です。

そこで、大口消費者の+5Vは、スイッチングを用いることにしました。その理由は、

- ① 効率がよい。これは入力電圧が高いほどよく、+12V用とトランスの巻線を共用するため、比較的高圧になるため、都合がよい。
- ② 効率がよいので、発熱が少なくコンパクトにできる。
- ③ 反面、
- ④ SWのスパイクが漏れて、ラジオなどにノイ

図1 スwitchング・レギュレータ部

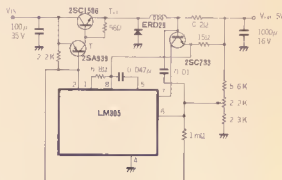
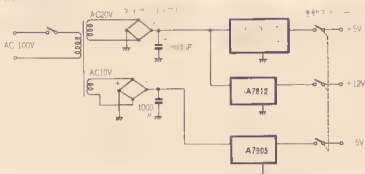




図2 全回路



ズを与えることがある。

② SWの周波数を低くすると、耳に聞こえるらしい。

③ 多少、入手しにくい部品がある。(ダイオード)

このように欠点もあるのですが、それは無視し、とにかくでっちあげたレギュレータが図1のもので、これにより、トランスは2巻線では+12、5V用、もう一つは、-5V用ということになります。市販品でこの目的にかなう巻線のトランスがなかった(2つのトランスを使えばあります。)ので、ジャンクでひろったオーディオ・アンプ用のトランスを巻き直して、ACで10V 1A、20V 4Aのトランスを作りました。筆者の場合は丁勤の巻線機が使えたため、もともとのトランスをバラすために手間をかけたほかは、たいした苦労はなかったのですが、普通はやはりできあいを買うか、特注になるでしょう。

## スイッチング・レギュレータ

チョップ型スイッチング・レギュレータを用い、入出力電圧、出力電流、リップル条件などから回路定数を決めました。動作原理、設計法は、他の文献を参考にしてもいい。ここでは作り方のコツを述べましょう。

まず、Lの値は計算式より得られますが、使うコアは、発振周波数が低くてもパルスを扱うわけですから、あまりロスのあるコアは不向きです。秋葉原でSFコイルと名づけられて売られているコイル(約200μHぐらいの値。ラフに作る時は適当)が適当でしょう。筆者は、こわれたスピーカー・ネットワーク用のコアを使用しています。

次に、ダイオードですが、通常の整流用ダイオードは使用できません。また、スイッチング用では、このぐらいの大電流が扱える石は普通入手が困難で、これ

がスイッチング電源製作時の最大のウイーク・ポイントになります。筆者の場合は、富士通のダイオードのサンプルセットが手元にあったので、その中から3AクラスのERD29を使いました。(ショートキーバリエ型で新電元などでも作っている。)

トランジスタは、スイッチング用のtOFF、つまりOFFになるための時間の短いパワー・トランジスタが必要です。ここでは、最近、f<sub>T</sub>10MHz以上のオーディオ用トランジスタが充分使用できます。ただ、速ければよいというものでもなく、高速といわれるRET(サブルーチンからのリターン命令でなく、リングエミッタ・トランジスタのこと)を使ったところ、スパイク分が大きくなりすぎてしまいました。これは、ほどほどの右スイッチング用といわれているものを使えば、ただし、電流容量に注意——まず、まちがいはないようです。注意点は、このぐらいでしょう。

## まとめ

上記のレギュレータとトランスに、+12、-5V用のレギュレータを加えて作りあげた電源の回路が図2です。通常、このタイプの電源を必要とするセトは、POWER ONについて制約がありますが、ここでは、AC側のスイッチ以外にDC側にも各々スイッチをつけ電源の投入は、同時にできるようにしています。

これで全部でき上がったつもりでいたのですが、1つ今後改良したい点があります。それは保護回路が各々独立しているため、試作中のCPUセトにつないだ場合、-5Vのみどこかでショートすると、そこだけが死んで残りが印加されたままになり、デバイスをこわすことが考えられることです。(特に8080、DCAMなど) 次には、これに対する処理も加え、グレード・アップしたいと考えています。







## NEW DEVICE

## INTEL 8086

Mr.1CHIP

昨年8085が発売になった時、それに前後して8008から続いたアーキテクチャの上位にあたるものを開発中というニュースが流れていました。しかし、昨秋にはそれが基本的に16ビットのデータバスを持つものということが明らかになっただけで、くわしいことはわかりませんでした。今年に入り、かなりこまかいことが発表されてきました。そこで、new CPU 8086がどのようなものか考えてみました<sup>1)</sup>。

## 8080/85と根本的に異なる点

まず基本的に16ビットを扱うCPUで、データバスは16ビットで作られています。しかし、8080の命令と上位コンパチビリティを考え、8ビットのデータの取り扱ひも容易になっています。つまり、今までの80の命令に、さらに16ビット系の命令が追加されています。

アドレッシングは1Mバイトまでに拡張されています。8080とのコンパチビリティを考えたからかも知れませんが、基本的にはCPU内部では64Kまで(つまり16ビット)のアドレッシングになっています。

割り込みは、くわしいことは不明ですが、メモリ中に256のエレメントをとり、割り込み処理のテーブルとして使用されるとのことです。

チップの構成は85ライクになっています。アドレスとデータの16ビットは共用していて、アドレス用にラッ

チが必要となっています。

クロックについては一歩後退でクロック・ジェネレータ8284の外付けが必要です。これにともない、専用のファミリーLS1が追加されています。

## ①クロック

クロック周波数は標準で5MHz、maxで8MHzが使用でき、メモリのサイクル・タイムは各々800ns、500nsです。そのためアクセスは各々480ns、295nsとなっており、5MHzのバージョンでは、現在安価な2102Aの450nsで充分です。これは8080から85に(タイミング的に)改良されたのを、さらに押し進め、アドレスが決まってからデータを必要とするまでの時間に余裕を持たせているようです。(5MHz=200nsを4クロックで1サイクルを終るようになっていいるとのことです。)

## ②レジスタ

8ビットの演算に加え、8ビットにおいてはビット

図2

RELOCATOR REGISTER FILE  
セグメントレジスタ

|    |               |
|----|---------------|
| CS | CODE SEGMENT  |
| DS | DATA SEGMENT  |
| SS | STACK SEGMENT |
| ES | EXTRA SEGMENT |

(各々16ビット)

図1 内部レジスタ

| レジスタ<br>8ビット |    |
|--------------|----|
| AH           | AL |
| BH           | BL |
| CH           | CL |
| DH           | DL |
| SP           |    |
| BP           |    |
| SI           |    |
| DI           |    |

A, B, C, Dの各レジスタは元来8ビットだったものが、各々8ビット×2, H, Lの2つにわかれ、16ビットとなっており8ビットの演算、16ビット演算両方に対応されています。

8080に對比すると、

Acc → Acc, さらに Acc が追加され、8/16ビット両用 Acc になる。

H, L → BHI, BH, BASE とよばれる。

B, C → CHI, CL, COUNT とよばれる。

D, E → DHI, DL, DATA とよばれる。

SP → SP, スタック・ポインタ (注 16ビット)

以上、Acc が16ビットになったのは8080と同じに使用するレジスタ群です。

BP → BASE POINTER

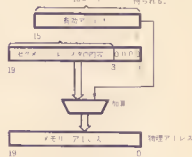
SI → SOURCE INDEX

DI → DESTINATION INDEX

の3つが新しいレジスタで、今までなかったインデックス・レジスタができたことになります。

以上のほかに、IP → PC Instruction Pointer, FLAG(H, L) → PSW Flag を含めて General Register File と呼ばれています。

物理アドレスの作成 インストラクション・ポインタ等より得られる。



命令・データを取り出すためのアドレスは上図のようにセグメント・レジスタの内容とPCの加算より得られ、またスタックを使用したときはSPとセグメント・レジスタを合わせて使用する様です。

このアドレッシングについては、自動割り込まれるか特別な命令によるかは不明です。

操作命令が追加されています。今の80でビットのテストなどを行なうにはAND、ORなどでマスクし、その結果をフラグでテストする方法がよく使われます。メモリもビット単位のセット・リセット機能を持っていないため、上記の手法と類似の考えでビット操作を行なっていましたが、これも可能なようです。さらに16ビットの演算(8080ではDAD命令でのみ16ビット演算が可能でした)が可能でしかも符号付で可能とのことでした。

さらに特筆すべきは、乗除算が8、16ビットどちらのモードでも可能になっていることです。(符号付とは、データが負数のときは2の補数表示となっている。)これらのため今までのAcc、B、C、D、E、H、Lのレジスタ群は図1のように変わっています。

図中の説明のようにBP、SI、DIの3つのレジスタがふえたこと、アキュムレータが2分割して使えることが今までと異なった点でしょう。これらのレジスタ類は新しく追加された命令、アドレッシングのために用意されているようです。これらのレジスタ群をインテルでは *Execution Unit* と呼んでいるようです。

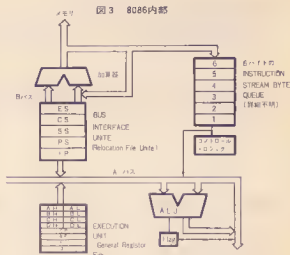
以上の外にも8086ではもう一組のレジスタ・ファイル群を持っています。(図2) これらは、まとめて *Relocation Register File* と名づけられており、その名のとおりリロケート可能なプログラムの実行、スタックが16ビットしかないための補助と、IPインストラクション・ポインタが16ビットしかないため、20ビットのアドレッシングを行なうためのメモリ・アドレスを作るために用意されています。

図3に上記レジスタ群以外のコンポーネントを含んだブロックを示しておきます。

### ③ 命令

詳細な情報はありますが、前に述べた加算・減算以外に乗除算のインストラクションを持っていること、さらにこれが8ビットのみでなく16ビット・データでも可能な点は、今までのマイコンどころでなくミニコンでも(ミニコン・レベルでは乗除算のソフト・パッケージを使うか、または乗除算用のハードの追加が必要)色々手間のかかったことがどのような形式のデータを扱うのかはわかりませんが、可能になっていることはやはり大革命的ともいえます。

また、命令の大きな変化としては、アドレッシングに関するものがあります。つまり、*Relocation Register File*、CS DS SS ES を利用したプログラムのリロケート可能なプログラムが容易に使えるようになったことです。6800は相対ジャンプ命令を多く持っていますが、それでも目的によっては完全にはリロケートな形のプログラムは作れませんでした。ただ、



8080/85命令体系をベースにしているため、どこまで自由にリロケートなプログラムが書けるか最終的なインストラクション・セットの発表が待たれるところです。(53年6月ころには発表になるのではと考えています。)

### ④ 雑感

80の命令をそのまま実行でき、さらに強力な演算命令群などが追加され、それが40ピンのLSIにつめられていることは、小型のコンピュータとしてはミニコンしかなかったころ、8080/8800といったワンチップCPUが発表された以上のおどろきと期待を持たざるを得ません。大きい見方をすれば、SSI→MSI→LSIと進んできたICの高集積化もいよいよLSI以上の集積度になるひとつの歴史の変革のときではないかと思えます。つまり、高密度な素子、高速化、その中への新しいアーキテクチャの追加(高密度化と同じ意味を持つ)、低消費電力など多くの面から見て興味深いものが、この8086にはまだまだあるようです。その別の面については、別の機会にできるとき説明できるように努力したいと思います。

### 【参考文献】

- 1) *ELECTRONICS誌*



# 8048のための 8035CPU



Mr. 1 CHIP

8085が発表されたのとは時期を同じくして1チップ8ビットマイコンとして、8048が発表されています。それ以来種々のおもしろい命令群、8085のファミリーチップを利用できる便利さにもかかわらず、ワンチップ指向ということからあまり注目されずにきました。ここでは、その命令をいろいろじくりまわす目的で、8048ファミリーである8035を用いたRAM（プログラム用）ベースの一種のトレーニングセットを試作してみました。

## 8048

まず、8048がどんなCPUか、ハード面について述べてみましょう。

●ワンチップCPU、つまり、プログラムROMもI/Oもワンチップにすべて詰めこまれています。ただ

### 8048/35ピン説明

|                                  |                                                                                       |
|----------------------------------|---------------------------------------------------------------------------------------|
| P <sub>10</sub> ~P <sub>17</sub> | I/Oポート1で8ビット入出力商用                                                                     |
| P <sub>20</sub> ~P <sub>27</sub> | I/Oポート2で8ビット入出力商用                                                                     |
| D <sub>0</sub> ~D <sub>7</sub>   | 両方向性のポートで外部に対しRD、WRの信号のタイミングに合わせて、データを扱う。外部プログラム使用時は下位8ビットのアドレスがでる。プログラムはPSENによりうけとる。 |
| T <sub>0</sub>                   | 状態をみる命令でテストできる入力ピンであり、クロック出力にも用いる。                                                    |
| T <sub>1</sub>                   | 命令によりテストできる入力ピン。イベント・カウンタの入力にも用いる。                                                    |
| INT                              | 割り込みの入力ピン (Low Active)                                                                |
| RD                               | 外部データメモリのリード・ストロープ用                                                                   |
| WR                               | 外部データメモリ用ライト・ストロープ                                                                    |
| ALE                              | 外部データ、プログラム・メモリのアドレス・ストロープ信号                                                          |
| PSEN                             | 外部プログラム・メモリよりのプログラム・フェッチ用信号                                                           |
| SS                               | シングル・ステップ実行用                                                                          |
| EA                               | 外部プログラム・メモリを使用するための切り換えピン                                                             |

し、試作用PROMの8748、プログラムROM外付けの8035、その他、ファミリーとして、I/Oピンを減らしたり、プログラム領域の大きさにより8021、8022などのファミリーチップがあります。

●内部のプログラムが不足のとき、外部に増設ができます。データRAMも外付けが可能です。

●I/Oピンが不足のときは外部にI/Oエクステンダー1Cがあり、8085のファミリーを使用すると、I/OのみでなくRAM、ROMも追加できます。

他のワンチップ型マイコンでも以上3点と同じような機能を持つものもありますが、データバスを直接外に出し、積極的に拡張を考えているワンチップCPUはありません。

I/Oポートはすべて入出力両方が可能で、8ビット2組あり、さらに1つ8ビットのバス・ポートをもっています。この他にハード面では、シングル・ステップ実行用のピン、外部プログラム・メモリを使用するためのピン（今回はこのピンを利用し、外部プログラムにRAMを利用しています。）その他に割り込み（命令によりマスク可能なものおよび内部のタイマを用いたもの）があります。この他にT<sub>0</sub>、T<sub>1</sub>という外部の状態を命令でテストする命令をハードと抱きあわせて持っています。この他にも色々普通の8ビットマイコンでは考えられない独特のものを持っています。これらソフトがらみの特色については、別にソフトの説明と合わせてしたいと思います。

### ピン配列図





## 外部プログラム・メモリ・タイミング



タイミングにしており、このときにRAMに入れるべきプログラムを書き込むことになります。

これらをまとめると、

- ① CPUは8048がマスク化したもので入丁不可能のため、8035を使用。
- ② プログラムはRAMを使用。

## リセット・シーケンス

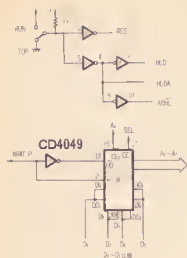
RSET端子がローレベルになると、

- ① プログラム・カウンタがゼロになる。
- ② サブルーチンスタック・ポインタがゼロになる。
- ③ レジスタ・バンク0がセレクトされる。
- ④ メモリ・バンク0がセレクトされる。
- ⑤ バスは高インピーダンスになる。
- ⑥ ポート1, 2は入力モードになる。
- ⑦ 割り込みは禁止される。
- ⑧ タイマを止める。
- ⑨ タイマ・フラグはクリアされる。
- ⑩ F0とF1のフラグはクリアされる。
- ⑪ T0よりのクロック・アウトが禁止される。

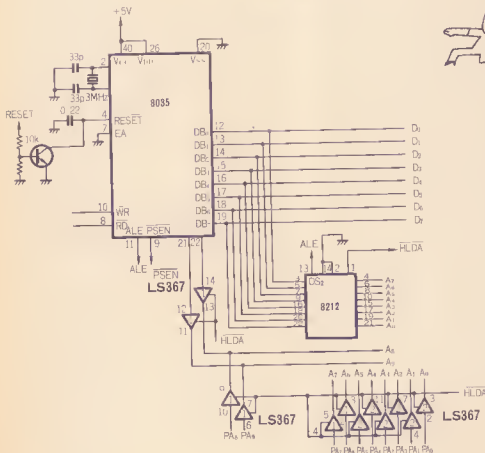
## 命令表

| 命令             | 説明                            | 1 | 2 | サブルーチン        | 説明                                 | 1 | 2 |
|----------------|-------------------------------|---|---|---------------|------------------------------------|---|---|
| ADD A, R       | Add register to A             | 1 | 1 | CALL          | Jump to subroutine                 | 2 | 2 |
| ADD A, @R      | Add data memory to A          | 1 | 1 | RET           | Return                             | 1 | 2 |
| ADD A, #data   | Add immediate to A            | 2 | 2 | RETR          | Return and restore status          | 1 | 2 |
| ADDG A, R      | Add register with carry       | 1 | 1 | CLR C         | Clear Carry                        | 1 | 1 |
| ADDG A, @R     | Add data memory with carry    | 1 | 1 | CPL C         | Complement Carry                   | 1 | 1 |
| ADDG A, #data  | Add immediate with carry      | 2 | 2 | CLR F0        | Clear Flag 0                       | 1 | 1 |
| ANL A, R       | And register to A             | 1 | 1 | CPL F0        | Complement Flag 0                  | 1 | 1 |
| ANL A, @R      | And data memory to A          | 1 | 1 | CPL F1        | Clear Flag 1                       | 1 | 1 |
| ANL A, #data   | And immediate to A            | 2 | 2 | CPL F1        | Complement Flag 1                  | 1 | 1 |
| ORL A, R       | Or register to A              | 1 | 1 | MOV A, R      | Move register to A                 | 1 | 1 |
| ORL A, @R      | Or data memory to A           | 1 | 1 | MOV A, @R     | Move data memory to A              | 1 | 1 |
| ORL A, #data   | Or immediate to A             | 2 | 2 | MOV A, #data  | Move immediate to A                | 2 | 2 |
| XRL A, R       | Exclusive or register to A    | 1 | 1 | MOV R, A      | Move A to register                 | 1 | 1 |
| XRL A, @R      | Exclusive or data memory to A | 1 | 1 | MOV @R, A     | Move A to data memory              | 1 | 1 |
| XRL A, #data   | Exclusive or immediate to A   | 2 | 2 | MOV R, #data  | Move immediate to register         | 2 | 2 |
| INC A          | Increment A                   | 1 | 1 | MOV @R, #data | Move immediate to data memory      | 2 | 2 |
| DEC A          | Decrement A                   | 1 | 1 | MOV A, PSW    | Move PSW to A                      | 1 | 1 |
| CLR A          | Clear A                       | 1 | 1 | MOV PSW, A    | Move A to PSW                      | 1 | 1 |
| CPL A          | Complement A                  | 1 | 1 | XCH A, R      | Exchange A and register            | 1 | 1 |
| DA A           | Decimal Adjust A              | 1 | 1 | XCH A, @R     | Exchange nibble of A and register  | 1 | 1 |
| SWAP A         | Swap nibbles of A             | 1 | 1 | XCH A, #data  | Exchange nibble of A and immediate | 1 | 1 |
| RLA            | Rotate A left                 | 1 | 1 | MOVX A, @R    | Move external data memory to A     | 1 | 2 |
| RLC A          | Rotate A left through carry   | 1 | 1 | MOVX @R, A    | Move A to external data memory     | 1 | 2 |
| RR A           | Rotate A right                | 1 | 1 | MOVP A, @A    | Move to A from current page        | 1 | 2 |
| RRC A          | Rotate A right through carry  | 1 | 1 | MOVP3 A, @A   | Move to A from Page 3              | 1 | 2 |
| IN A, P        | Input port to A               | 1 | 2 | MOV A, T      | Read Timer/Counter                 | 1 | 1 |
| OUTL P, A      | Output A to port              | 1 | 2 | MOV T, A      | Load Timer/Counter                 | 1 | 1 |
| ANL P, #data   | And immediate to port         | 2 | 2 | START T       | Start Timer                        | 1 | 1 |
| ORL P, #data   | Or immediate to port          | 2 | 2 | START CNT     | Start Counter                      | 1 | 1 |
| INS A, BUS     | Input BUS to A                | 1 | 2 | STOP TONT     | Stop Timer/Counter                 | 1 | 1 |
| OUTL BUS, A    | Output A to BUS               | 1 | 2 | EN TONT       | Enable Timer/Counter Interrupt     | 1 | 1 |
| ANL BUS, #data | And immediate to BUS          | 2 | 2 | DIS TONT      | Disable Timer/Counter Interrupt    | 1 | 1 |
| ORL BUS, #data | Or immediate to BUS           | 2 | 2 | EN I          | Enable external interrupt          | 1 | 1 |
| MOVD A, P      | Input Expander port to A      | 1 | 2 | DIS I         | Disable external interrupt         | 1 | 1 |
| MOVD P, A      | Output A to Expander port     | 1 | 2 | SEL RB0       | Select register bank 0             | 1 | 1 |
| ANLD P, A      | And A to Expander port        | 1 | 2 | SEL RB1       | Select register bank 1             | 1 | 1 |
| ORLD P, A      | Or A to Expander port         | 1 | 2 | SEL MB0       | Select memory bank 0               | 1 | 1 |
| INC R          | Increment register            | 1 | 1 | SEL MB1       | Select memory bank 1               | 1 | 1 |
| INC @R         | Increment data memory         | 1 | 1 | ENTD CLK      | Enable Clock output on T0          | 1 | 1 |
| DEC R          | Decrement register            | 1 | 1 | NOP           | No Operation                       | 1 | 1 |
| JMP add        | Jump unconditional            | 2 | 2 |               |                                    |   |   |
| JMP @A         | Jump indirect                 | 1 | 2 |               |                                    |   |   |
| DJNZ R, add    | Decrement register and skip   | 2 | 2 |               |                                    |   |   |
| JC add         | Jump on Carry = 1             | 2 | 2 |               |                                    |   |   |
| JNC add        | Jump on Carry = 0             | 2 | 2 |               |                                    |   |   |
| JZ add         | Jump on A Zero                | 2 | 2 |               |                                    |   |   |
| JNZ add        | Jump on A not Zero            | 2 | 2 |               |                                    |   |   |
| JTO add        | Jump on T0 = 1                | 2 | 2 |               |                                    |   |   |
| JT0 add        | Jump on T0 = 0                | 2 | 2 |               |                                    |   |   |
| JT1 add        | Jump on T1 = 1                | 2 | 2 |               |                                    |   |   |
| JT1 add        | Jump on T1 = 0                | 2 | 2 |               |                                    |   |   |
| JFO add        | Jump on F0 = 1                | 2 | 2 |               |                                    |   |   |
| JF1 add        | Jump on F1 = 1                | 2 | 2 |               |                                    |   |   |
| JTF add        | Jump on timer flag            | 2 | 2 |               |                                    |   |   |
| JNI add        | Jump on INT = 0               | 2 | 2 |               |                                    |   |   |
| JBS add        | Jump on Accumulator Bit       | 2 | 2 |               |                                    |   |   |





# CD4016



●そのためDMA的な状態を強引に作る。  
この3つのアクションをしていることになります。  
さて、実際に使用しているデバイスは、CPU8035、RAM2101×2、アドレス・ラッチ8212。

他に、アドレス用にトライステート・バッファLS367を使用。基板に余白がまだあるため、8155をI/O拡張、データRAM用として追加していますが、まだボード上にはスペースがあり、今後色々な機能が追加できそうです。

## 動作法

詳しいタイミング等による動作中の信号の動きの大体は回路図を参照してもらうことにして、ここでは、いわゆるスイッチなどのとり扱いのみについて説明しましょう。今まで8080、6800などの簡単なシステムを作った方なら、プログラムをメモリに入れるためにCPUをDMAが可能な状態にして、1アドレスごとにスイッチをバチバチして入れたことがあると思います。

本機でも、“ローダ”なるプログラムも、固定したI/Oも持たないため……というより、そのようなプログラムを作るためのトレーニング・セットであるから、当然持っていない。DMAを行ないたいのですが、本来DMA機能を持たないため、先に述べたようにDMA化する回路を作りました。図でもわかるように8080などではプログラムの実行の途中でもDMAが可能でしたが、本機ではRESET状態でしかバスをあげる事ができません。これはCPUの設計方針がまったく異なるためで、どうしようもないのです。(といいたいのですが実は8048/35自体の使い方がる程度決まれば多少の外付けのデータメモリを利用するなどして8080のようなDMA機能を持たせることができるようで、このセットにより、プログラミングのこつをのみこんでから実験を予定しています。)

多少脱線しましたがhiノとにかくRESET/HOLD兼用のスイッチを入れると、自動的にパネルのスイッチ・レジスタとプログラム・メモリはダイレクトに接続され、アドレス・スイッチでアドレスを指定し、データ・スイッチを設定し、DEPスイッチを押してやるだけで操作としてこれ以外のことはありません。

このとき、8080などにあったように、プログラムの入れそこない……アセンブル時のミス、プログラムのローケーションのミスなど……により、プログラムが暴走し、それ自体をこわすことは絶対ありません。

なぜなら、プログラム用メモリとデータ用のメモリは完全に分かれていて、当然のことながらプログラム領域のメモリに対してCPUは書き込む命令を持っていないからです。万が一、こわれたときはプログラム書き込み回路の不良動作しか考えられません。ですが

ら自殺プログラムを作ってしまうかとビクビクしながらCPUを使うといった心配の必要はありません。

さて、プログラムの自殺がないことがわかったのですが小さなプログラムを実験的に作り、そのチェックはメモリの中のデータ・エリアにその結果を命令により書き込むようなプログラムを追加しておき、実行後そのエリアを開けて見るということで小さなシステムのプログラムの作製を行なうことが多いと思います。残念ながら本システム(?)ではデータ・メモリはCPUの内部にあり強引に開けて見ることはできません。そのため上記のようなプログラム・チェック法は不可能で何か別な方法が必要で。

①CPU自体がI/Oピンを持つので、このI/Oピンの先にLEDをつけておき、I/O命令により結果を表示するようにする。つまり、モニタ・プログラムでなくモニタ・サブルーチンを作る。このとき、8048はシングル・ステップ動作がハードでできる以外に割り込みおよび疑似的な割り込みを実行できるので、これを利用して“デバッグがもどき”ができる。

②バス上にLATCHを設けCPUからのWR信号で結果をLATCHに書き込み、LATCHの内容は①と同じくLEDなどを表示させる。動作は①と同様の方法でデバッグにも用いることができる。

③データ専用のメモリをプログラムメモリと同様の方法でパネルよりアクセスできる位置に配置する。これによると入力データもメモリに入れるため①、②のようにデータをイミディエートの形で、いちいち入れる手間がなくなります。また③の考えを発展させ、まったく8080と類似のシステムとすることも可能でしょう。

## まとめ

苦しまぎれに“変な手”は使っているものの、これ当初の目的であるワンチップ・マイコン8048のプログラミング・トレーニング用のセットができたわけです。このセットではCPUが本来持てるプログラム・メモリは4Kワードまでのところ、I/Oピンを悪用し、バンク切り換えに用い、4K以上に増設でき、データRAMも同じように増設できるなど、ハード的にいろいろくふうして大きくする楽しみがまだ多く残されています。つまり、ワンチップ型ではありながらマルチチップ型マイコンのように自分自身の唯一無二のシステムを作り上げる楽しみがあります。8048シリーズはそんなマイコンといえるでしょう。



# 丸善洋書売場案内

## ●パターン解析

Pattern Analysis. Lectures in pattern theory.  
Vol. 11. Applied Mathematical Science, Vol. 24)  
By U. Grenander and L. H. Lailou. '78. (Springer-  
V., Berlin)  
〈近著〉…………… 元価 ¥ 4,800

## ●アルゴリズムの一般理論

An Introduction to the General Theory of  
Algorithms. (Theory of Computation Ser. By  
M. Machtey and P. Young. Thomond Books)  
〈近著〉…………… 元価 ¥ 6,180

## ●自動プログラミングの論理学

Studies in Automatic Programming Logic.  
(Artificial Intelligence Ser. By Z. Manna.  
Thomond Books)  
〈近著〉…………… 元価 ¥ 2,770

## ●デジタル信号コンピュータとプロセッサ

Digital Signal Computers and Processors.  
(IEEE Pr. Selected Reprint Ser.) Ed. by A. C.  
Salazar. '77. (Wiley, New York)  
〈近著〉…………… 元価 ¥ 4,010

問い合わせ先 (03) 272-7211

# I/O ソフトウェア・サービス開始

★I/Oでは地方にいて、マイコンのソフトウェアの入手が思うままに行かない方のために、ソフトウェア・サービスを開始します。

## ●SWTPC用

|                  |                            |
|------------------|----------------------------|
| Space Voyage     | ¥12,000(送料込) [カセット、解説リスト付] |
| Battleship       | ¥12,000( * ) [カセット、解説リスト付] |
| Klimgon Capture  | ¥12,000( * ) [カセット、解説リスト付] |
| Text Editor      | ¥12,000( * ) [カセット、解説リスト付] |
| Editor/Assembler | ¥12,000( * ) [カセット]        |
| 8K BASIC         | ¥12,000( * ) [カセット]        |
| 4K BASIC         | ¥ 8,000( * ) [カセット]        |

## ■新着 (SWTPC)

株式売買ゲーム ¥12,000(送料込) [カセット、解説リスト付]  
ソフトウェア・パッケージ No. 3 ¥20,000  
[1枚切りゲームのプログラムリストと解説]

●6800用2K BASIC ¥3,000(送料 ¥500) [紙テープ、解説付]

●6502用2K BASIC ¥3,000(送料 ¥500) [紙テープ、解説付]  
¥3,300(送料 ¥500) [KIM用カセット、解説付]

## (代金送料先)

●現金書留または郵便振替 (東京5 22510) で

〒151 東京都渋谷区代々木2-5-1 羽田ビル507

株式会社 ソフトウェア係

■必要なプログラム名を明記してください。

(例) SWTPC用 Battleship

# New Products

## §ディスプレイ/キーボード用 コントロールIC§

■MTX-A-1は、カナダ・マトロックス社から発表されたLEDディスプレイ/キーボードのコントロールIC。内部構成は、CPUとのインターフェイス・ロジック、キャラクター・ジェネレータ用マスクROM (5×7×64ビット)、ディスプレイ用メモリ (32×8ビット)、タイミング回路、コントロール・ユニットからなっている。

### 〈特徴〉

▶5×7ドット・マトリクスLEDを32個までドライブできる。▶64接点までのキーボードをスキャンする入出力ポートを持つ。▶8ビットのデータバスを持ち、CPUに対しI/Oまたはメモリとして用いることができる。▶CPUからディスプレイ・メモリの内容をリード、ライト可能▶22線の8ビット・インストラクション (クリア・ディスプレイ、

ローテート・ディスプレイ、インクリメント・カーソルなど) を受け、実行する。

▶40ピン N-MOS 1C

〈価格〉¥16,300 (1~24個のサンプル価格)

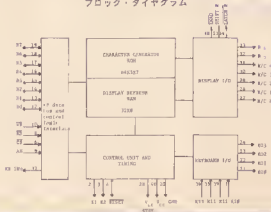
〈問い合わせ先〉インテリニクス㈱

〒160 東京都新宿区西新宿7-4-7

☎ (03) 369-1101



ブロック・ダイアグラム





## 初級 特訓講座

## PART3

BASIC を  
始めよう!

佐藤雅春

## サブ ルーチンと関数

## プログラムのループ

前回FOR-NEXTを使ったプログラムのループを説明しました。

```
FOR I=0 TO 9
  :
NEXT I
```

というループでは、変数Iをループカウンタ（何回くり返したかを覚えておくための変数）に使っています。

そして、Iは必ず1ずつずつ増加していきます。まずIは0、NEXT Iまで

プログラム5

LIST

```
500 PRINT "NUMBER : N",
510 PRINT TAB(17), "NA2",
520 PRINT TAB(29), "NA3"
525 PRINT
530 FOR I=0 TO 1 STEP 0.1
540 PRINT TAB(3), I,
550 PRINT TAB(16), I^2,
560 PRINT TAB(28), I^3
570 NEXT I
READY
RUN
```

| NUMBER : N | NA2 | NA3  |
|------------|-----|------|
| 0          | 0   | 0    |
| .1         | .01 | .001 |
| .2         | .04 | .008 |
| .3         | .09 | .027 |
| .4         | .16 | .064 |
| .5         | .25 | .125 |
| .6         | .36 | .216 |
| .7         | .49 | .343 |
| .8         | .64 | .512 |
| .9         | .81 | .729 |
| 1          | 1   | 1    |

の間のプログラムを実行します。NEXTのある行までくると、次にIは1になって再びFORの次のステートメントを処理します。これはIが9になるまで自動的にくり返されます。

0から1まで0.1ステップずつの数の2乗、3乗の表を作ってみましょう。

```
FOR I=0 TO 10
PRINT(I/10)^2,(I/10)^3
NEXT I
```

というプログラムでもよいのですが、

```
FOR I=0 TO 1 STEP 0.1
PRINT I^2,I^3
NEXT I
```

1行目のSTEP 0.1というところが前と違い、そのためにループカウンタは0.1ずつ増加され、1になるまでループを回ります。

プログラム5とその実行結果をみてください。

一般にステップの値は負でもよく、ここに式を入れることもできます。

例1

```
10 FOR J=0 TO 9
20 :
30 IF M=N THEN 60
40 :
50 NEXT J
60 :
70 IF P<0 THEN 10
```

```
FOR I=J TO K STEP
(K-J)/10
```

とするとJからKまでの値を10ステップに分割してループを回ってきます。もちろんK<Jであってもかまいません。

## ぜひ知っておきたいこと

FOR-NEXTループを使う上での注意としては、

- ①ループの中でループカウンタの値を勝手に変えてはいけません。
- ②正しいサブルーチン・コール(GOSUB, 後で説明します。)は無条件に許されます。
- ③GOTOは条件のあるなしにかかわらず使えますが、そのループの中に再び帰って来て、FOR-NEXTをいずれは処理し終るようではなくてはなりません。つまり、例1のようなプログラムはエラーとなる場合があります。特に、終了してなかったループを再びその先頭から実行するような場合は、内部スタックが混乱してしまいます。

ループの仕上がりとして、階乗の計算をするプログラムを作ってみましょう。整数Nの階乗は次の式で与えられます。

$$R = N \cdot (N-1) \cdot (N-2) \cdot \dots \cdot 1$$

ある値にある数乗し、その結果に別の値を乗し、その最後の結果だけ

が必要なのでから、

# プログラム6

```
R=1
FOR I=N TO 1 STEP -1
R=R*I
NEXT I
```

とすると上の式通りの順序となりますが、乗算は交換則が成り立ち、1を乗じても結果は変わらないので、整理するとプログラム6のようになります。なお、このプログラムの実行を中断するにはCTRL/Cなどを使わなければなりません。

## サブルーチン

もし、いくつかの値の階乗を計算し、その結果を使ってさらに演算をするような場合はどうしたらよいでしょう。

```
NI/R/(R*(N-R))
(これは階乗を表わす記号です)
```

この式では、3つの数の階乗を計算しなくてはなりません。したがって、階乗の計算をするループを3つ書かなくてはならない……と思いますか？

- ①Aの階乗を計算して結果をBに代入するプログラムを作っておく。
- ②A=Nとし、①のプログラムを動かしてBを得て、NI=Bとする。
- ③A=Rとして同様に、①のプログラムでBを得て、RI=Bとする。
- ④A=N-Rとし、①のプログラムによりBを得る。
- ⑤NI/(RI\*B)を計算する。

このようなステップを続ければループは1つで済みます。ステップ②では①のプログラムを呼び出した後NI=Bとし、ステップ③に移りました。ステップ④ではA=Rとした後①のプログラムを呼び出し、今度はRI=Bとしました。

これをGOTOステートメントで処理するのではやっかいです。なぜかというと①のプログラムはそれが終了した後に往く先が場合によって異なるからです。BASIC言語ではこのような場面に使うための特別なステートメントGOSUBとRETURNが用意されています。

```
10 GOSUB 100
20
30 GOSUB 100
```

## LIST

```
600 INPUT "NUMBER? ",N
610 IF N=INT(N) THEN 640
620 PRINT "INTEGER PLEASE!"
630 GOTO 600
640 R=1
650 FOR I=2 TO N
660 R=R*I
670 NEXT I
680 PRINT TAB(12);N," " " " = ",R
690 GOTO 600
READY
RUN
```

```
NUMBER? 2      2 ! = 2
NUMBER? 3      3 ! = 6
NUMBER? 4      4 ! = 24
NUMBER? 5      5 ! = 120
NUMBER? 20     20 ! = 2.432902E+18
NUMBER? 25     25 ! = 1.551121E+25
NUMBER? 33     33 ! = 8.6833174E+36
NUMBER?
STOP IN LINE 600
READY
```

```
40
50 END
100
120 RETURN
```

このようなプログラムは、10→100→120→20→30→100→120→40→50という順に処理されます。つまりGOSUBのあとに書いてあるラインナンバーの行に飛び、RETURNがあるとそのGOSUBが書いてあった行の次の行に帰ってくるのです。

GOTOではどこから来たのかという情報は失われてしまいましたが、GOSUBはそれを覚えているのが大きな違いです。

①から⑤までのステップをBASIC言語で書き表わしたのがプログラム

## ▶マイコンでチェス



写真提供: ESPラボラトリー

## LIST

```
700 INPUT "N ? ",N
705 INPUT "R ? ",R
710 A=N
715 GOSUB 770
720 NI=B
725 A=R
730 GOSUB 770
735 RI=B
740 A=N-R
745 GOSUB 770
750 S=NI/(RI*B)
755 PRINT TAB(12);N," " "C",R," " = ",S
760 PRINT S
765 END
770 B=1
775 FOR I=2 TO A
780 B=B*I
785 NEXT I
790 RETURN
READY
RUN
```

```
N ? 3
R ? 1      3 C 1 = 3
READY
RUN
```

```
N ? 4
R ? 2      4 C 2 = 6
READY
RUN
```

```
N ? 8
R ? 4      8 C 4 = 70
READY
RUN
```

```
N ? 12
R ? 6      12 C 6 = 924
READY
```

7です。ナンバー770-790の間がサブルーチンで、コンピュータのプログラミングでは非常によく使われる手法です。

## プログラム 8

## LIST

```

800 PRINT "DEGREE", TAB(14), "SINE", TAB(30), "COSINE", TAB(46), "TANGENT"
810 PRINT
820 P1=3, 14.159
830 FOR D=0 TO 85 STEP 5
840 R=2*P1*(D/360)
850 PRINT D, TAB(12), SIN(R), TAB(28), COS(R), TAB(44), SIN(R)/COS(R)
860 NEXT D
READY
RUN

```

| DEGREE | SINE      | COSINE        | TANGENT   |
|--------|-----------|---------------|-----------|
| 0      | 0         | 1             | 0         |
| 5      | .08715567 | .9961947      | .08748859 |
| 10     | .17364803 | .9848078      | .17632682 |
| 15     | .25881983 | .9659259      | .26794895 |
| 20     | .34201987 | .9396927      | .36396991 |
| 25     | .42261792 | .9063079      | .46630722 |
| 30     | .49999961 | .8660256      | .57734969 |
| 35     | .57357601 | .81915234     | .70020676 |
| 40     | .64278715 | .76604482     | .83909862 |
| 45     | .70710631 | .70710723     | .9999987  |
| 50     | .76604398 | .64278814     | 1.1917519 |
| 55     | .81915159 | .57357706     | 1.4281457 |
| 60     | .866025   | .50000071     | 1.7320475 |
| 65     | .9063074  | .42261907     | 2.1445019 |
| 70     | .9396923  | .34202113     | 2.7474686 |
| 75     | .9659255  | .25882012     | 3.732034  |
| 80     | .9848076  | .17364933     | 5.6712433 |
| 85     | .9961946  | 8.7156975E-02 | 11.42989  |

## LIST

```

800 PRINT "DEGREE", TAB(16), "SINE", TAB(32), "COSINE", TAB(48), "TANGENT"
810 PRINT
820 P1=3, 14.159
830 FOR D=0 TO 85 STEP 5
840 R=2*P1*(D/360)
850 PRINT D,
852 PRINT TAB(12), X12F8, SIN(R),
854 PRINT TAB(28), X12F8, COS(R),
856 PRINT TAB(44), X12F8, SIN(R)/COS(R)
860 NEXT D
READY
RUN

```

| DEGREE | SINE      | COSINE    | TANGENT   |
|--------|-----------|-----------|-----------|
| 0      | 0         | 1         | 0         |
| 5      | .08715567 | .9961947  | .08748859 |
| 10     | .17364803 | .9848078  | .17632682 |
| 15     | .25881983 | .9659259  | .26794895 |
| 20     | .34201987 | .9396927  | .36396991 |
| 25     | .42261792 | .9063079  | .46630722 |
| 30     | .49999961 | .8660256  | .57734969 |
| 35     | .57357601 | .81915234 | .70020676 |
| 40     | .64278715 | .76604482 | .83909862 |
| 45     | .70710631 | .70710723 | .9999987  |
| 50     | .76604398 | .64278814 | 1.1917519 |
| 55     | .81915159 | .57357706 | 1.4281457 |
| 60     | .866025   | .50000071 | 1.7320475 |
| 65     | .9063074  | .42261907 | 2.1445019 |
| 70     | .9396923  | .34202113 | 2.7474686 |
| 75     | .9659255  | .25882012 | 3.732034  |
| 80     | .9848076  | .17364933 | 5.6712433 |
| 85     | .9961946  | .08715698 | 11.42989  |

READY

## エラーとネスト

GOSUBのあとに書いたナンバーの行が実際にない場合は、ラインナンバーエラーに、GOSUBなしにRETURNがあった場合は、それを示すメッセージがでてきます。

サブルーチンの中で、さらに別のサブルーチンを呼び出すことも可能です。これをネスティングと言います。何重にネスティングできるかは、

RAMの容量によって制限されるものが多いようです。

あるサブルーチンが、プログラムの中の別な場所から2回以上呼び出されない場合は、それをあえてサブルーチンとする必要はなくGOTOステートメントを使って、あるいは順序を入れ替えるだけで処理できるはずですが、しかし実際のプログラムでは、1回しか呼ばれなくてもサブルーチン形式にすることがあります。これはプログラムの作成、デバッグ

時にその方が便利だからです。

プログラムを作ろうとするとき、一般には、まずフローチャートを書きますが、その各ステップをサブルーチンにしておくと各部分の機能の単純化ができて見通しもよくなり、エラーも起こしにくいでしょう。そして、最後にそれぞれのルーチンを順序よく結びつけるプログラムを作れば全体が完成します。デバッグ時にも各ルーチンの機能が単純化されていますから、比較的容易にエラーも見えてきます。

## ファンクション

BASICでは数値的な関数のうち、あるものは特に定義しなくても利用できます。関数の性質上ある値を与えると結果も値として得られます。これを変数に代入したり、あるいは直接PRINTさせることができます。たとえば、

B=SQR(A)

とすると、Aの値(Aの2乗根)がBに代入されます。三角関数のSIN, COS, TAN, また指数関数、対数関数であるEXP, LOGなども同様に使えます。ほかにはABS(絶対値), |A|, INT(その値を超えない最大の整数)といったものも利用できます。これらは、

D=(SQR(A)+SIN(B))/C

などというように式の中に自由に書くことができます。なお三角関数の角度はラジアンで与え、指数、対数関数の底はeとなっているのが普通ですが、それぞれ角度、底を10と指定可能なBASICもあります。これはBASICがもともと科学技術用の言語であるためそうなっているのです。

## 三角関数表を作ろう

角度は0-85°まで、5°ステップでSIN, COS, TANの3つの値を横に並べて書くことにします。360°は2πラジアンですから、D度をラジアンに変換するには2π(D/360)を計算すればよいでしょう。

FOR D=0 TO 85 STEP 5

```
R=2*3.14159*(D/360)
PRINT SIN(R), COS(R),
NEXT D      TAN(R)
```

というプログラムでもよいのですが、これでは何の値がどれなのかはつきりしません。プログラム8を見てください。これを実行した結果はRUN以降です。少し見やすくなりましたが小数点の位置が一定していま

せん。これを揃えるには、フォーマット・アウトプットを活用すると容易です。これを使った結果をその次に示しておきますが、フォーマット・アウトプットについては回を改めてご紹介したいと思います。

### 注意

ここで使用したBASICの、PRINT

ステートメントは、普通用いる□と□を区別して使うことができません。すべて□を使い、機能は通常の□と同じです。フォーマットは、TABと特別なステートメントを使うようになっています。このようなBASICはどちらかというと少数派ですが、ご不信に思われるむきもあると感じましたのでおことわりしておきます。

## BASICの品詞分類

### コマンド

BASIC言語のプログラムが動いていない状態で使用するもので、次のようなものです。

```
LIST RUN NEW
LOAD SAVE CON
```

原則としては、プログラムの中にライン・ナンバーをつけて書くものではないのですが、ものによってはあるコマンドを実行してくれる場合もあります。

### ステートメント

プログラムの中で用いるもので、「ある操作が行なわれる」と理解してよいでしょう。

```
LET PRINT GOTO
IF-THEN INPUT GOSUB
RETURN READ DATA
RESTORE FOR-TOSTOP
NEXT STOP END
DIM REM PEEK
OUT CALL
```

といったところが代表的なもので、いわばBASIC言語の主役たちです。

### ファンクション

やはりプログラムの中で用いるもので、「ある値が得られるもの」と理解してよいでしょう。

```
INT ABS SQR
LOG EXP SIN
COS TAN RND
PEEK IN
```

多くのファンクションは、アーギュメント(引数)を必要とします。例えばINT(A)など。引数をカッコの中に入れて与えます。結果は値として得られます。それを変数に代入したり、式式の中に入れたり、また直接PRINTさせることもできます。



### オペレータ

これは3つに分けられます。

#### ●アリスティック・オペレータ

数理的な演算を指定する記号で、

```
*, / (加減乗除)
↑ (アップアロー、またはシフト
レックス、累乗を表す)
```

以上、5種が、数に用いられます。

これは2つの引数を前後に書いて使うファンクションと考えることもできるでしょう。なぜこのように分かれてしまったかというと、BASIC言語の表記をなるべく我々が通常使っているものと一致させようとしたからです。

#### ●リレーショナル・オペレータ

の6つでIF-THENステートメントの中で大小関係の判断に用いられます。FOR, LETに使う□は、しいてリレーショナル・オペレータと考えずに、単なる表記上の記号と思った方がよいでしょう。

#### ●ロジカル・オペレータ

論理演算—ブール代数の演算ではありません。

```
AND OR NOT
```

以上3つが通常使えます。エクスクルージブORの入っているBASICもあります。

これらはリレーショナル・オペレータとともにIF-THENの中で用いられます。

IF A=B AND C=D THEN...  
要するにこれは英語そのものです。A=B, C=Dが同時に成立したときに限って、THEN以下のことが実行されます。OR, XORも通常の論理の定義と同じです。

NOTはユニナリ・マイナスと同様な働きをし、オペレータの優先順位は次の通りです。

NOT, unary minus (-)

```
↑
*, /
+, -
~, <, >, <=, >=
AND
OR
```

### ダイレクト・ステートメント

さきほどステートメントの説明をしたところで、「プログラムの中で」と書きましたが、プログラムが走っていない状態、つまり、コマンド・レベルでも多くのステートメントを実行できます。これをダイレクトステートメントと呼び、ダイレクトに使えないものの方が数が少ない。

```
RETURN NEXT DATA
```

などはその便でない代表です。もともと、BASICによってはもっと制限されている場合もあるので、くわしくは実際にやってみるか、マニュアルをご覧ください。

マルチプルステートメント/ライン (1行に複数のステートメントを書くこと)を許しているBASICの多くは、その機能を生かせばFOR-NEXTをダイレクトに処理してくれます。

この用途は、何といってもデバッグ用です。また、ダイレクトに開く、オペレータも使えるので、

```
PRINT 32768+8192+4
```

などを書くことも簡単代わりにもなります。

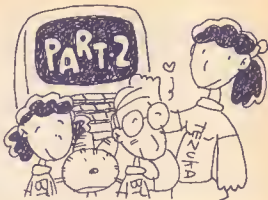
### その他

このほか、BASIC言語にとつて意味のあるものは、

- ①誤りを知らせてくれるエラーメッセージ。これを人間が使ったことはできませんが、
- ②エディティングに使うラインナンバーとコントロール・キャラクター (CTRL, コンソール・キーを押しながら使うアンプリタブル・キャラクター)
- ③パワースイッチ。これはコンピュータに対して人間が使える最後のエラーメッセージです。

# BASIC で遊ぼう!

〈PART 2〉 手塚佐知 (コンピュータ・ラブ)



## ディメンションとストリング

SAN JOSEの展示会は3月3日～5日の間にぎやかに開かれました。今年は寒くて、しかも雨が異常に多かったCALIFORNIAの気候は、あまりこういうことに準備のない地方に水害やガケくずれをひきおこしました。SAN JOSEも毎日雨でしたが、そのために緑が例年になく美しかったようです。会場は大変な混雑ぶりでした。日本の方々も多く、いろいろ手に入りくいソフトウェアなどを買っていました。大物のコンピュータをお買いになった方は、日本で買うより安い」ということでしたけれど、故障したらどうするのかなどと考えながら3日間充分に見学で過ごしました。来年はぜひ、皆様も行かれたらいかがですか?

### 1. 添字付変数とディメンション

変数を効率よく使うためにはどんなことが考えられているでしょうか。例えば、次々と多くの変数を使う場合もあれば、特定の変数を選び出した場合、これまでの一般的な変数では系統的にはできないのです。親が自分の子供を呼ぶのに、長男なら一郎、3番目なら三郎というような方式です。10人もいたら案外ですよ。

変数にも実は、A(1), A(2), A(5), A(10)などというやり方をとってあらわすのです。このような変数を添字付変数とよんでいます。この添字付変数を使うためには、変数の数をはじめに宣言しないといけません。さもないと連続した記憶場所がとれず、うまくいかなくなってしまいます。

この宣言は、DIMというインストラクションを使います。

```
10 REM SOEJITSUKI HENSU
```

```
20 REM OHKISA NO SENGEN
30 DIM A(50)
40 FOR K=0 TO 50
50 INPUT A(K)
60 NEXT K
70 PRINT
80 FOR K=0 TO 50
90 PRINT A(K),
100 NEXT K
999 END
```

30は、AがA(0)～A(49)までの50個の添字付変数であることを宣言しています。50は、Kを指標として、入力データをA(0)～A(49)までしようことを意味します。90はこの逆で、A(0)～A(49)を印字していくわけです。いかがですか、ずいぶん気楽にできるでしょう?

### 2. \$マークはストリングのサイン

ドル安で、なかなか大変な経済界ですが、BASICの\$はあまり円には関係ありません。実はこのドルマーク、BASICではストリング変数の印なのです。いままで変数は数値をとり扱ってきたわけですが、文字を扱えるという面白いことができます。

```
10 LET A$ = "U.S.$"
20 LET B$ = "YEN"
30 INPUT P
40 LET Q=1:P
50 PRINT
60 PRINT
70 PRINT A$: " "; P: B$
80 PRINT B$: " "; Q: A$
90 END
```

ワリント文の中で変数扱いOK

行番号10, 20を見てください。A\$, B\$というように変数名の最後はドルマークです。この場合、右辺には“(ダブルクォート)”でかこんだストリングを置くことができます。つまりストリングを変数として扱うことが可能になったのです。ですから、70, 80のところで普通の変数と同じようにとり扱われています。30, 40のP, Qは数値を扱う変数で、A\$, B\$とはまったく別のものです。

```

RUN
?250          PRINT
               PRINT

```

```

U. S. $=250YEN
YEN 0.004U. S. $

```

このストリング変数は、ディメンション指定をする必要のあるBASICもありますから注意しましょう。DIMはまだ覚えてばかりですから、すぐおわかりになるとと思います。

```
5 DIM A$(10), B$(3)
```

などとやっておけばよいのです。もっと大きなストリングを扱いたいときは、カッコの中の数をその分だけ増してください。普通は255文字までが扱えます。

ところで、これではあまり面白くはありません。もっとうまく使えないだろうか、ということを考えてみましょう。

### 3. INPUTも\$マークで

何かこのタイトルは貯金するみたいですね。独身貴族はじゃんじゃん貯金が増えて、夢のAPPLE-IIが買えるかも……。

```

5 DIM A$(10), B$(10)
10 INPUT A$, B$
20 INPUT P
30 PRINT
40 PRINT A$, "-": P; B$
50 PRINT B$, " ": 1; P; A$
60 END

```

```

RUN
?U. S. $, YEN
?250

```

```

U. S. $=250YEN
YEN 0.04U. S. $

```

行番号10はストリング変数の入力です。INPUTはこれまでキーボードから数値を入れる目的で使われていましたが、ここでは文字列(ストリング)が自由に入られます。ということは数値と同様変更も可能になりますし、一般の変数と同様に比較などもできそ



うです。

ここで1つ集計のプログラムでも作ってみましょう。項目は5としてみます。

```

5 REM SHUKEI PROGRAM
10 DIM GOKEI(5)
15 DIM KOMOKUS(8)
20 DIM Q$(1)
30 FOR K 0 TO 4
35 LET GOKEI(K) 0
40 NEXT K
50 INPUT KOMOKUS$, VALUE,
60 IF KOMOKUS$ "UFO" THEN
  K=0
70 IF KOMOKUS$ "KLINGON"
  THEN K 1
80 IF KOMOKUS$ "STAR" TH
  EN K 2
90 IF KOMOKUS$ "OBAQ" TH
  EN K-3
100 IF KOMOKUS$ "TORP" T
  HEN K 4
110 LET GOKEI(K) GOKEI(K)
  +VALUE
120 PRINT "OWARI(Y N)";
130 INPUT QS
140 IF QS N THEN GOTO 50
150 PRINT
160 PRINT "UFO", "KLINGON",
  "STAR", "OBAQ", "TORP"
170 FOR K 0 TO 4
180 PRINT GOKEI(K),
190 NEXT K
200 PRINT
999 END

```

```

RUN
OBAQ      5      OWARI(Y N) N
KLINGON   3      OWARI(Y N) N
UFO       20     OWARI(Y N) N
OBAQ      3      OWARI(Y N) N
TORP      15     OWARI(Y N) N
STAR      120    OWARI(Y N) Y

```

| UFO | KLINGON | STAR | OBAQ | TORP |
|-----|---------|------|------|------|
| 20  | 3       | 120  | 8    | 15   |

KOMOKUSはUFO, KLINGON, STAR, OBAQ, TORPの5種を用います。行番号10は、GOKEIをしまう場所をGOKEI(0)~GOKEI(4)までの5つとしています。行番号15はKOMOKUSの文字列は最大8つまでと限定しています。20はYESまたはNOのためのもので、YまたはNの1文字のためのものです。

行番号50はKOMOKUSとその値を入力するもので、60~100で項目号のGOKEIの格納場所を選んでいくわけです。つまりKの値がここできまります。次に110番で合計を計算して格納しておきます。

行番号120はデータの終了かどうかをチェックするルーチンで、ここにもQ\$なる文字変数が使われていることには目してください。

FOR...NEXTのループが2度も出ていることは、もうお気づきのことと思います。

これでは合計だけが最後にあらわれませんから、中間をみたいときはこれではうまくありません。

家計簿や小づかい帳はどうしたらよいのでしょうか。必要な項目としては日付、費目、金額、残額があります。日付と費目をストリング変数、金額、残額は数値変数とすると、下のようなことになるでしょう。

| HIZUKE | HIMOKU  | KINGAKU | ZANGAKU |
|--------|---------|---------|---------|
| 4 20   | TOSHO   | 350     | 19200   |
| 4 21   | KONETSU | 1500    | 17700   |

こうなるとすべてを記憶させておかなければなりませんので、どんな方法をとるかで大分考え方が変わってきます。4 20とかTOSHOなどというのはストリングとして、しまっておくのかどうかという点で、いささか\$マークでできるかどうか心配ですね。金額や残額のところはK(N), Z(N)ということで、簡単そうです。

皆さん大いに頭の体操のためにも、独身の方々は来米のおよめさんのためにも(コンピュータによる家計簿付なんてどんなものでしょう)ぜひ、がんばってみてください。

## New Products

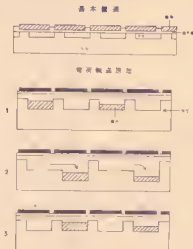
### §BBD素子を使った

### エコー・ミキサー§

■MX-550は、ソニーがこのほど開発した1,024段BBD(Bucket Brigade Device)ICを採用したソリッド・ステート・エコー・ミキサー。アクティブエコー・ミキサーで、すべてのチャンネルにエコーが付加できる。カラオケ演奏はもちろん、シンセサイザを使う効果的なサウンドづくりに適している。同時に姉妹機種としてMX-250が発表されている。

(特徴)

▶MOS FETを用いたBBD ICをエコー回路に2個採用。絶電方式エコー回路のため、外部磁界、振動に影響されず、S/N比、ひずみの面で改善されている。▶エコー・タイムは、スイッチにより3段階の遅延時間(約40, 64, 128ms)が選べ、好みのエコー効果が得られる。

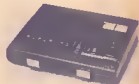


フォノの各入力端子を装備。▶出力は2チャンネル。各チャンネル入力をL, R, Mに切り換えるスイッチを持ち、ステレオ・ミキシングが可能▶エコーレベル可変ボリューム付▶3電源方式▶持ち運びに便利な把手つき薄型ケース。

CX-765



MX-550



MX-250



《価格》MX-550 ¥49,800  
MX-250 ¥29,800

《問い合わせ先》ソニー株式会社

〒141 東京都品川区北品川6-7-35

☎(03)448-2111



# TK-80BS

## 入門 ③

# BSのステートメント

戸塚 文男 (Bit-INN)

前編にひきつづいて、TK-80BSで使うステートメントをプログラム例を交えて解説します。さらにBASICを理解するために、インタープリタの構造について説明を加えていきましょう。

### 各ステートメント(文)の説明

#### 〔3〕POKE文

一般の1つにPOKE[変数1], [変数2]というのがある。[変数1]で示される16進絶対番地に「変数2」の16進データを格納します。

#### 〔例1〕POKE A, B

BASICインタープリタのワークエリアの中の変数エリア (8526H番地～8559H番地、図3参照) に格納されている変数Aのアドレスに変数Bのデータを格納します。

POKE A, Bの動作

|         |   |     |       |
|---------|---|-----|-------|
| 8 5 2 6 | A | 0 0 | 下位バイト |
| 8 5 2 7 |   | 8 8 | 上位バイト |
| 8 5 2 8 | B | 3 0 |       |
| 8 5 2 9 |   | 0 0 |       |
| }       | } | }   |       |
| 8 8 0 0 | - | 3 0 |       |

実行後

〔例2〕POKE 8900H, 80H ..... (正)  
POKE D, E ..... (正)  
POKE 8500H, E ..... (誤)  
POKE C, 30H ..... (誤)

上のように変数と16進のアドレスまたはデータを組み合わせることは出来ません。

#### 〔4〕IF文

基本形としてIF[関係式][文]で表われ、関係式の条件が成立すれば継続する文を実行し、不成立であれば次の行へ進みます。

〔例1〕10 IF A>B+1 LET A=0  
20 LET B=B\*2

A>B+1という条件が成立すれば、A=0として



から次の文へ進みます。不成立であれば、LET A=0を無視して行番号20へ進みます。

〔例2〕100 IF A GOTO 200  
110 A=A-1  
200 LET A=B\*3

Aが真数であれば200番へジャンプします。違う場合は110番へ進みます。

#### 〔5〕GOTO文

基本形としてGOTO[行番号]で無条件にGOTOの後に書かれている行へジャンプします。ただし、ジャンプ先の行番号がプログラム内になければエラーになります。

100 IF A=0 GOTO 300  
300 PRINT "OK"

10 GOTO A+100

Aに代入された数に100を加えた行番号へジャンプする。

#### 〔6〕STOP文

プログラムの翻訳実行を中止します。

100 LET A=10  
110 PRINT A  
200 STOP

#### 〔7〕FOR...NEXT文

FOR n1 TO n2 STEP m

n1: 初期値 [変数]=[変数, 定数または式]

n2: 制限値 [変数, 定数または式]

m: 増分値 [変数, 定数または式]

反復実行文でNEXTまでのループを、n1からn2ずつ変えてn2になるまで反復実行します。

10 FOR A=1 TO 10 STEP 1  
20 PRINT A  
30 NEXT A



この例は変数Aを1, 2, 3...10とループさせて、1から10までを1ステップずつPRINTさせるプログラムです。

この場合STEP数が1の場合は、STEPを省略できます。

```

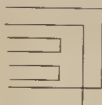
FOR A = 1 TO 10
  FOR B = 10 TO 1
    :
  10 NEXT B
310 NEXT A
  
```

Nest  
構造

1つのFOR・NEXTのループの中に、別のFOR・NEXTのループを組み入れることができます。ただし、この場合には必ずNest構造でなければなりません。



(正) Nest構造に  
なっている。



(誤) Nest構造に  
なっていない。

#### FOR・NEXT文を使うときの注意事項

ここでの説明しているレベル-1 BASICに限っては、このループの中からループ外のサブルーチンなどへジャンプして、復帰するということは禁止されています。

もし、これを行なった場合には、プログラムを破壊するときがあるので、絶対使わないでください。

#### (8) GOSUB文

基本形としてGOSUB[行番号]でBASICプログラム内のサブルーチンへジャンプします。サブルーチン内で処理が終了しますと、RETURNコマンドによって、サブルーチンをコールした次の行番号から実行します。

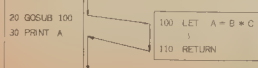
```

10 LET B = 10
GOSUB 100
PRINT A
100 LET A = B * C
110 RETURN
  
```

この例ではB=10、C=20と代入したあと、行番号100のAを求めるサブルーチンへジャンプし、110のRETURNコマンドによって、30へジャンプしてAの結果をPRINTします。

メインプログラム

サブルーチン



BASIC内のサブルーチンがメインプログラムに直接結ばれている場合は、メインプログラムで使っている変数はサブルーチン内でも全く同じ意味を持ちます。

#### (9) RETRUN文

BASICのサブルーチン・プログラムの終りに入れるステートメントです。サブルーチンをコールした次の行番号へジャンプして実行。

#### (10) CALL文

基本形としてCALL[16進絶対番地]Hマシン・ランゲージで作成されたサブルーチンへの分岐命令文で、与えられた16進絶対番地へ分岐します。

例1 50 CALL B200H  
60 LET A = B + 1

機械語サブルーチン

BASICプログラム

```

50 CALL B200H
60 LET A = B + 1
  
```

B200H

```

PUSH B
PUSH D
PUSH H
PUSH PSW
POP PSW
POP H
POP D
POP B
RET
  
```

#### (11) PRINT文

T V画面に印刷要素をディスプレイします。

##### ① 改行

```

PRINT "          "
10 PRINT: PRINT: PRINT:
10 PRINT,
20 PRINT: 3行改行します
10 PRINT:
  
```

##### ② 変数値、計算値出力

```

10 PRINT
9文字分のエリ
10 PRINT ABS(A)
9文字分のエリアにAの値を出力
  
```

##### ③ メッセージ出力

10 PRINT "BASIC PROGRAM"  
これを実行しますと引用符号"..."で囲んだメッセージ文をPRINTします。ただし、引用符号の中に引用符号を入れることはできません。

##### ④ 出力形式の指定

出力形式の指定は、おおよび面によって行なわれます。

PRINT

ここで変数の値をA=50、B=580、C=2800として

実行すると、

```

|-----9文字分-----|-----9文字分-----|-----5文字分-----|
UUUUUUUU50, UUUUUUU580, U5800

```

数値の出力は、図による桁数の指定がなければ、つねに9文字分のエリアを保持します。

#### ●任意パターン、文字出力

基本形としては、PRINT H [コード], [コード] 文了、パターンなどのキャラクタ・ジェネレータ内のコードを16進数で指定すると、キャラクタを出力します。

```
10 PRINT #100 05, 09
```

を実行すると、A、B、Cと表示します。

このプリント文では、カーソルの自動修正をしていないのでカーソル位置の修正を行なわないと、表示されたキャラクタが破壊されてしまいます。

#### [12] INPUT文

基本形としては、INPUT[変数i], ... [変数n] です。プログラムを実行させ、INPUT文までくると図を出力し、変数の値を入力するまで待ちます。

```

1 INPUT A, B,
A, B, Cの を区切ります。
2 INPUT "メッセー" A

```

引用符号の中のメッセージを出力し、次にA?とディスプレイされるので、Aの値をキーボードから入力します。

#### [13] CLEAR文

TV画面の全画面のキャラクタをスペースでクリアした後に、カーソルポイントをイニシャライズ(X=1, Y=1)してから、図マークを出力します。

```
10 CLEAR
```

#### [14] CURSOR文

基本形はCURSOR [定数1], [定数2]

CURSOR [変数1], [変数2]

カーソルポイントを任意の位置に移動します。

TV画面の構成は、

X方向(横方向) 1~32 (定数1, 変数1で指定)

Y方向(縦方向) 1~16 (定数2, 変数2で指定)

ただし、指定された値が最大値(X=32, Y=16)

以上の場合には最大値を指定します。

#### □関数

レベル-1 BASICでは、次の4つの関数を使用できます。

- (a) ABS (絶対値)
- (b) RND (擬似乱数)
- (c) SIZE (メモリ容量)
- (d) PEEK (代入)
- [a] ABS

基本形はABS ([変数, 定数または式])

( )内の値の絶対値を求めます。

```

10 PRINT ABS (-5)
PRINT B=ABS(A+10)
PRINT A=ABS(Y)

```

#### [b] RND

基本形はRND ([変数, 定数, 式])

1~( )内で与えられる値の範囲で、整数の乱数を発生します。

```
1 LET A=RND(100)
```

1から10までの乱数を発生\*レベル-1の乱数発生は規則性がある、ゲームなどに应用するにはあまり実用的ではありませんが、8524Hと8525の乱数格納アドレスを工夫すれば改善されます。

#### [c] SIZE

BASICプログラム・エリアの残りのバイト数を求めます。

```
10 PRINT SIZE
```

#### [d] PEEK

基本形はPEEK ([16進絶対アドレスまたは変数])  
で、16進絶対アドレスの内容、または変数で与えられるアドレスの内容を求めます。

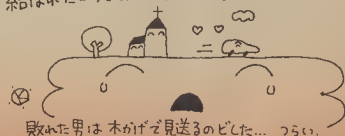
```
10 INPUT A: PEEK (8200H)
```

8200番地の内容を変数エリアAに入れます。

```
10 LET A=PEEK (B)
```

Bの値が16進で8200とすると8200番地の内容を変数エリアに格納します。

結ばれた二人 ホルシェにのっていくのだした



# BASICインタープリタ構造

BASICインタプリタの中には、

(a)イニシャライズ・ルーチン

(b)入力ルーチン

(c)編集ルーチン

(d)翻訳実行ルーチン

以上4つの基本ルーチンに分類されています。そのフローチャートを図1に示します。

## [1] 各ルーチンの説明

### (a)イニシャライズ・ルーチン

イニシャライズ・ルーチン(1)

①キーボード入力によるプログラム・エリア最終アドレスの初期設定

②TV画面の初期化

イニシャライズ・ルーチン(2)

①スタック・ポインタの初期設定

②BASICワークエリア、フラグの初期設定。

### (b)入力ルーチン

入力ルーチンは、キーボードより「**修正**」の入力があるまで入力して、初めに行番号があれば**編集ルーチン**へ、ダイ

図1 BASICインタープリタのフローチャート

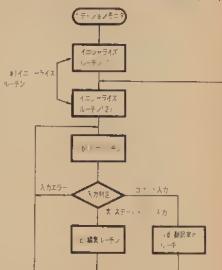


図3 BSシステム・テーブルおよびインタープリタのワークエリア

|      |                                           |  |
|------|-------------------------------------------|--|
| 8400 | Eレジスタ                                     |  |
| 8401 | Dレジスタ                                     |  |
| 8402 | Cレジスタ                                     |  |
| 8403 | Bレジスタ                                     |  |
| 8404 | フラグ                                       |  |
| 8405 | Aレジスタ                                     |  |
| 8406 | Lレジスタ                                     |  |
| 8407 | Hレジスタ                                     |  |
| 8408 | プログラム・カウンタL                               |  |
| 8409 | プログラム・カウンタH                               |  |
| 840A | スタック・ポインタL                                |  |
| 840B | スタック・ポインタH                                |  |
| 840C | モードセマフォア 00=システムプログラム起動中 01=ユーザープログラム起動中  |  |
| 840D | システムセマフォア 0F=モニタ通過モード F0=モニタ読み込みモード       |  |
| 840E | BPAアドレス { 下位バイト                           |  |
| 840F | LPアドレス { 上位バイト                            |  |
| 8410 | LPカウンタ (Bin)                              |  |
| 8411 |                                           |  |
| 8412 | 読み込み発生時                                   |  |
| 8413 | PC表示用エリア                                  |  |
| 8414 |                                           |  |
| 8415 |                                           |  |
| 8416 | コマンド第1パラメータ                               |  |
| 8417 | 編集エリア (Bin)                               |  |
| 8418 | コマンド第2パラメータ                               |  |
| 8419 | 編集エリア (Bin)                               |  |
| 841A | コマンド第3パラメータ                               |  |
| 841B | 編集エリア (Bin)                               |  |
| 841C | 入力文字数 (Bin)                               |  |
| 841D | キーボード入力バフ・オーバーフロー (CHAR) { 最大80文字+CR LF } |  |
| 841E |                                           |  |
| 841F |                                           |  |
| 8420 | テープ書き込み開始アドレス                             |  |
| 8421 |                                           |  |
| 8422 | テープ書き込み終了アドレス                             |  |
| 8423 | Bin→ASC IIコンバート・ルーチン                      |  |
| 8424 | キャラクタ格納エリア (CHAR)                         |  |
| 8425 |                                           |  |
| 8426 | テープ読み込み開始アドレス                             |  |
| 8427 |                                           |  |
| 8428 | テープ読み込み終了アドレス                             |  |

|      |                             |  |
|------|-----------------------------|--|
| 8479 | 出力文字格納エリア (CHAR)            |  |
| 847A | 出力メッセージ文字数 (Bin)            |  |
| 847B | 出力メッセージ                     |  |
| 847C | 光頭アドレス                      |  |
| 847D | カーソル・ポインタX (3~32)           |  |
| 847E | カーソル・ポインタY (1~16)           |  |
| 847F | ビデオRAM                      |  |
| 8480 | アドレス格納エリア                   |  |
| 8481 | 読み込み発生時ユーザー・シェルフ情報 (107バイト) |  |
| 8482 |                             |  |
| 8483 | BASIC用ワーク (データ処理バフ)         |  |
| 8484 |                             |  |
| 8485 | BASIC実行時ワークエリア              |  |
| 8486 |                             |  |
| 8487 | 実行時ワークエリア { 下位バイト           |  |
| 8488 | 実行時ワークエリア { 上位バイト           |  |
| 8489 | 実行時ワークエリア { 下位バイト           |  |
| 848A | 実行時ワークエリア { 上位バイト           |  |
| 848B | BASIC実行時ワークエリア              |  |
| 848C |                             |  |
| 848D |                             |  |
| 848E |                             |  |
| 848F |                             |  |
| 8490 |                             |  |
| 8491 |                             |  |
| 8492 |                             |  |
| 8493 |                             |  |
| 8494 |                             |  |
| 8495 |                             |  |
| 8496 |                             |  |
| 8497 |                             |  |
| 8498 |                             |  |
| 8499 |                             |  |
| 849A |                             |  |
| 849B |                             |  |
| 849C |                             |  |
| 849D |                             |  |
| 849E |                             |  |
| 849F |                             |  |
| 84A0 |                             |  |
| 84A1 |                             |  |
| 84A2 |                             |  |
| 84A3 |                             |  |
| 84A4 |                             |  |
| 84A5 |                             |  |
| 84A6 |                             |  |
| 84A7 |                             |  |
| 84A8 |                             |  |
| 84A9 |                             |  |
| 84AA |                             |  |
| 84AB |                             |  |
| 84AC |                             |  |
| 84AD |                             |  |
| 84AE |                             |  |
| 84AF |                             |  |
| 84B0 |                             |  |
| 84B1 |                             |  |
| 84B2 |                             |  |
| 84B3 |                             |  |
| 84B4 |                             |  |
| 84B5 |                             |  |
| 84B6 |                             |  |
| 84B7 |                             |  |
| 84B8 |                             |  |
| 84B9 |                             |  |
| 84BA |                             |  |
| 84BB |                             |  |
| 84BC |                             |  |
| 84BD |                             |  |
| 84BE |                             |  |
| 84BF |                             |  |
| 84C0 |                             |  |
| 84C1 |                             |  |
| 84C2 |                             |  |
| 84C3 |                             |  |
| 84C4 |                             |  |
| 84C5 |                             |  |
| 84C6 |                             |  |
| 84C7 |                             |  |
| 84C8 |                             |  |
| 84C9 |                             |  |
| 84CA |                             |  |
| 84CB |                             |  |
| 84CC |                             |  |
| 84CD |                             |  |
| 84CE |                             |  |
| 84CF |                             |  |
| 84D0 |                             |  |
| 84D1 |                             |  |
| 84D2 |                             |  |
| 84D3 |                             |  |
| 84D4 |                             |  |
| 84D5 |                             |  |
| 84D6 |                             |  |
| 84D7 |                             |  |
| 84D8 |                             |  |
| 84D9 |                             |  |
| 84DA |                             |  |
| 84DB |                             |  |
| 84DC |                             |  |
| 84DD |                             |  |
| 84DE |                             |  |
| 84DF |                             |  |
| 84E0 |                             |  |
| 84E1 |                             |  |
| 84E2 |                             |  |
| 84E3 |                             |  |
| 84E4 |                             |  |
| 84E5 |                             |  |
| 84E6 |                             |  |
| 84E7 |                             |  |
| 84E8 |                             |  |
| 84E9 |                             |  |
| 84EA |                             |  |
| 84EB |                             |  |
| 84EC |                             |  |
| 84ED |                             |  |
| 84EE |                             |  |
| 84EF |                             |  |
| 84F0 |                             |  |
| 84F1 |                             |  |
| 84F2 |                             |  |
| 84F3 |                             |  |
| 84F4 |                             |  |
| 84F5 |                             |  |
| 84F6 |                             |  |
| 84F7 |                             |  |
| 84F8 |                             |  |
| 84F9 |                             |  |
| 84FA |                             |  |
| 84FB |                             |  |
| 84FC |                             |  |
| 84FD |                             |  |
| 84FE |                             |  |
| 84FF |                             |  |

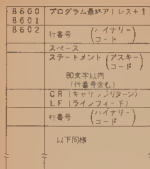
1. 実行出力ルーチン・パラメータブロック  
メッセージ出力ルーチン  
パラメータ・ブロック

ビデオRAM・アドレス計算  
ルーチン・パラメータ・ブロック

ただけよ  
さらば  
ぶっこめけん...



図2 プログラムエリアおよびフォーマット



レクト実行の場合は翻訳実行ルーチンへジャンプします。  
エラーがある場合はエラーメッセージをビデオラムに表示して  
エニシャリズルーチン(2)へジャンプします。

### (c) 編集ルーチン

入力ルーチンでバッファリングされた、行番号、ステートメントを8602番地からのプログラム・エリアに、次の3つの処理をリストアップします。(図2)

- ①削除 行番号+「**DELE**」または行番号+「**DEL**」+「**RE**」  
の場合は、指定した行を削除します。ただし、該当する行番号がない場合はエラーになります。
- ②挿入 入力した行番号がエリア内にはない場合は、メモリ容量のオーバーフロー・チェックを行ない、格納できればエリア内の他の行番号との大小の比較を行ないながら、行番号の小さい順に格納します。
- ③文換 エリア内に書き込んだ行番号と同じ行番号を入力すると、メモリのオーバーフロー・チェックを行ない、格納できるときは前の行番号とステートメントを削除してから、

TK-80BSの入出力特性

### (1) 入力信号

| ピン番号<br>(100P<br>32ピコタ) | 信号名             | 負荷の許容値  |
|-------------------------|-----------------|---------|
| A10~A17                 | AB15~AB7        | 1 TTL以下 |
| B10~B17                 | AB7~AB0         | 1 TTL以下 |
| A27                     | HOLDA+RESET     | 1 TTL   |
| B20                     | MEMR            | 3 TTL以下 |
| B21                     | MEMW            | 6 TTL以下 |
| B26~B33                 | DB7~DB0(入力モード時) | 1 TTL以下 |
| B44                     | RESET           | 1 TTL以下 |

### (2) 出力信号

| ピン番号<br>(100P<br>32ピコタ) | 信号名          | ドライブ能力 |
|-------------------------|--------------|--------|
| B6                      | SDK 80       |        |
| B8                      | +            |        |
| B19                     | TK-80RAMセレクト | 9 TTL  |
| B24                     | TK-80MONセレクト | 10 TTL |
| B26~B33                 | DB7~DB0      | 9 TTL  |
| B35                     | EXTINT       | 6 TTL  |

入力した行を挿入します。

### (d) 実行ルーチン

コマンドは、LIST, RUN, NEW, TAPEH, LOADHの5つがあります。このルーチンでは指定したコマンドとこの5つのコマンドと一致するものがあれば、各実行ルーチンのエントリアドレスにジャンプします。

### [2] BASICワークエリア

図3にBASICインタプリタのワークエリアおよびモニターのシステム・テーブルのマップを表示します。

## TK-80BS用プリンタについて

### ① ソースリストのハードコピー

#### ① プリンタ (ハード・コピー) の必要性

② プログラム・デバックをやり易くする。長いプログラムをデバックする場合、ビデオRAMでは全リストを表示することが不可能、必要な行番号付近のリストを毎回ディスプレイしなくてはならない、そこで全部のリストを一度に見る必要があります。

③ 最終リストをファイルしておく場合、コーディング・シートなどよりも確実です。

④ 結果やデータのファイルにも便利

⑤ ダイレクト・メールの送達の際、ファイルである住所、氏名などをプリントすれば手間がはぶける。

#### ② PR 40Z (サウスウエスト) の仕様

##### ① プリンタ部

文字構成: 5×7ドット・マトリクス

文字種類: 英・数字・特殊記号… (標準)

カナ… (オプション)

文字数: 40キャラクタ/行

印字速度: 75行/分 (50Hzの時)

印字用紙: 98mm幅

印字領域: 最大85mm (可変)

##### ② インターフェイス

0010 REM 30行 200

0020 IF=1000: A=3

0030 C

0040 F.J=3T030

0050 F.J=3T010

0060 D.J=J

0070 P.J=0

0080 NEXT J

0090 NEXT I

0100 F.J=117012

0110 G99:8820

0120 NEXT I

0130 F.J=217022

0140 G99:8820

0150 NEXT I

0160 G99:8820

0170 D.J=1

0180 P.J=3000: 200 \*\*\*

0190 D.J=12

0200 P.J=3000: 200 \*\*\*

0210 D.J=14

0220 IN "77777777"

0230 IF M=0: G.250

0240 G99:8820

0250 G.210

0260 F.J=1704

0270 A=0: G99:8820 (4)

0280 NEXT I

0290 G99:8820 (4)

0300 NEXT I

0310 IF A=0: G.360

0320 IF A=0: G.350

0330 IF B=0: G.350

0340 S=0

0350 G.360

0360 S=2

0370 G.360

0380 IF B=0: G.375

0390 G.350

0400 S=0

0410 S=5

0420 G.410

0430 G.410

0440 G.410

0450 G.410

0460 G.410

0470 G.410

0480 F.J=10300

0490 A=0 (4)

0500 NEXT I

0510 M=0: M=5: K=0: S=0

0520 IF M=0: G.900

0530 IF M=0: G.950

0540 F.J=121031

0550 J=2

0560 D.J=J

0570 NEXT I

0580 G99:8820

0590 G.160

入力: TTLコンパチブル

出力: TTLコンパチブル

●電源 AC 100V 50/60Hz 0.9A (印字中)

## ●TK-80BSとの接続

①I/OポートはTK-80基板上の8255のPBポートおよびPC<sub>0</sub>, PC<sub>1</sub>, PC<sub>4</sub>からダイレクトに接続するだけで、I/Fを必要としない。

②現在開発されているソフトは、次の3つの機能を持っている。

## ●ソースリストのハードコピー

BASICで組んだプログラム・リストをプリントする。

## ●メモリ・ダンプのハードコピー

メモリの内容をヘキサ形式でプリントする。

●現在、TV画面にディスプレイしてあるものをそのままプリントする。

●このコントロール・リフトはTK-80のμPD454の0-2に書き込んでコントロールします。

PR-40Zの外観



## ②メモリ・ダンプの例

|      |    |    |    |    |    |    |    |    |
|------|----|----|----|----|----|----|----|----|
| F000 | F3 | 31 | 00 | 86 | 3E | C3 | 21 | 25 |
| F008 | F1 | 32 | 00 | 83 | 22 | 0E | 83 | 32 |
| F010 | F0 | 13 | 22 | FE | 13 | 32 | 08 | 20 |
| F018 | 22 | 09 | 20 | 3E | B2 | 32 | FF | 70 |
| F020 | 97 | 32 | FF | 70 | 3E | 03 | 32 | FF |
| F028 | 70 | 3E | 0F | 32 | F9 | 70 | 3A | F8 |
| F030 | 70 | 00 | 6C | FA | 3E | 21 | 32 | 7A |
| F038 | 84 | 21 | 5F | F0 | 22 | 78 | 84 | 00 |
| F040 | 52 | FA | 21 | 00 | 84 | 97 | 06 | 0A |
| F048 | 77 | 23 | 05 | C2 | 48 | F0 | 21 | 00 |
| F050 | 8C | 22 | 0A | 84 | 97 | 32 | 10 | 84 |
| F058 | 97 | 32 | 0C | 84 | C3 | 80 | F0 | 2A |
| F060 | 2A | 20 | 42 | 41 | 53 | 49 | 43 | 20 |
| F068 | 53 | 54 | 41 | 54 | 49 | 4F | 4E | 20 |
| F070 | 40 | 4F | 4E | 49 | 54 | 4F | 52 | 20 |
| F078 | 56 | 31 | 2E | 30 | 20 | 2A | 00 | 20 |
| F080 | F3 | 3E | 0F | 32 | 00 | 84 | 3E | 24 |

```

0800 P=400-9
0801 F,I=PT09
0802 F,I=4T09
0803 O,I,I,J
0804 P,H80
0805 NEXT J
0806 NEXT I
0807 IF P=29 GOTO 6,810
0808 P=P+10;GOTO 10
0809 GOTO 1
0810 RETURN
0811 F,I=3T010
0812 O,I,I,J
0813 P,H80
0814 NEXT J
0815 RETURN
0816 F,I=12T030
0817 J=14
0818 O,I,I,J
0819 P,H80
0820 NEXT J
0821 NEXT I
0822 F,I=3T031
0823 O,I,I,J
0824 P,H80
0825 NEXT I
0826 F=5
0827 F=5
0828 IF G=4 P,H80+C3+C2+C3
0829 IF G=1 P,H80+B2+B4+B0
0830 IF G=2 P,H80+D2+D3+D0
0831 IF G=3 P,H80+P2+P4+P0
0832 GOTO 1
0833 O,I,I,F
0834 F,I=1
0835 IF G=1 P,H80+D2+B4+B0
0836 IF G=1 P,H80+D2+B4+B0
0837 IF G=1 P,H80+D2+B4+B0
0838 IF G=1 P,H80+D2+B4+B0
0839 IF G=1 P,H80+D2+B4+B0
0840 IF G=1 P,H80+D2+B4+B0
0841 IF G=1 P,H80+D2+B4+B0
0842 IF G=1 P,H80+D2+B4+B0
0843 IF G=1 P,H80+D2+B4+B0
0844 IF G=1 P,H80+D2+B4+B0
0845 IF G=1 P,H80+D2+B4+B0
0846 IF G=1 P,H80+D2+B4+B0
0847 IF G=1 P,H80+D2+B4+B0
0848 IF G=1 P,H80+D2+B4+B0
0849 IF G=1 P,H80+D2+B4+B0
0850 IF G=1 P,H80+D2+B4+B0
0851 IF G=1 P,H80+D2+B4+B0
0852 IF G=1 P,H80+D2+B4+B0
0853 IF G=1 P,H80+D2+B4+B0
0854 IF G=1 P,H80+D2+B4+B0
0855 IF G=1 P,H80+D2+B4+B0
0856 IF G=1 P,H80+D2+B4+B0
0857 IF G=1 P,H80+D2+B4+B0
0858 IF G=1 P,H80+D2+B4+B0
0859 IF G=1 P,H80+D2+B4+B0
0860 IF G=1 P,H80+D2+B4+B0
0861 IF G=1 P,H80+D2+B4+B0
0862 IF G=1 P,H80+D2+B4+B0
0863 IF G=1 P,H80+D2+B4+B0
0864 IF G=1 P,H80+D2+B4+B0
0865 IF G=1 P,H80+D2+B4+B0
0866 IF G=1 P,H80+D2+B4+B0
0867 IF G=1 P,H80+D2+B4+B0
0868 IF G=1 P,H80+D2+B4+B0
0869 IF G=1 P,H80+D2+B4+B0
0870 IF G=1 P,H80+D2+B4+B0
0871 IF G=1 P,H80+D2+B4+B0
0872 IF G=1 P,H80+D2+B4+B0
0873 IF G=1 P,H80+D2+B4+B0
0874 IF G=1 P,H80+D2+B4+B0
0875 IF G=1 P,H80+D2+B4+B0
0876 IF G=1 P,H80+D2+B4+B0
0877 IF G=1 P,H80+D2+B4+B0
0878 IF G=1 P,H80+D2+B4+B0
0879 IF G=1 P,H80+D2+B4+B0
0880 IF G=1 P,H80+D2+B4+B0
0881 IF G=1 P,H80+D2+B4+B0
0882 IF G=1 P,H80+D2+B4+B0
0883 IF G=1 P,H80+D2+B4+B0
0884 IF G=1 P,H80+D2+B4+B0
0885 IF G=1 P,H80+D2+B4+B0
0886 IF G=1 P,H80+D2+B4+B0
0887 IF G=1 P,H80+D2+B4+B0
0888 IF G=1 P,H80+D2+B4+B0
0889 IF G=1 P,H80+D2+B4+B0
0890 IF G=1 P,H80+D2+B4+B0
0891 IF G=1 P,H80+D2+B4+B0
0892 IF G=1 P,H80+D2+B4+B0
0893 IF G=1 P,H80+D2+B4+B0
0894 IF G=1 P,H80+D2+B4+B0
0895 IF G=1 P,H80+D2+B4+B0
0896 IF G=1 P,H80+D2+B4+B0
0897 IF G=1 P,H80+D2+B4+B0
0898 IF G=1 P,H80+D2+B4+B0
0899 IF G=1 P,H80+D2+B4+B0
0900 IF G=1 P,H80+D2+B4+B0
0901 IF G=1 P,H80+D2+B4+B0
0902 IF G=1 P,H80+D2+B4+B0
0903 IF G=1 P,H80+D2+B4+B0
0904 IF G=1 P,H80+D2+B4+B0
0905 IF G=1 P,H80+D2+B4+B0
0906 IF G=1 P,H80+D2+B4+B0
0907 IF G=1 P,H80+D2+B4+B0
0908 IF G=1 P,H80+D2+B4+B0
0909 IF G=1 P,H80+D2+B4+B0
0910 IF G=1 P,H80+D2+B4+B0
0911 IF G=1 P,H80+D2+B4+B0
0912 IF G=1 P,H80+D2+B4+B0
0913 IF G=1 P,H80+D2+B4+B0
0914 IF G=1 P,H80+D2+B4+B0
0915 IF G=1 P,H80+D2+B4+B0
0916 IF G=1 P,H80+D2+B4+B0
0917 IF G=1 P,H80+D2+B4+B0
0918 IF G=1 P,H80+D2+B4+B0
0919 IF G=1 P,H80+D2+B4+B0
0920 IF G=1 P,H80+D2+B4+B0
0921 IF G=1 P,H80+D2+B4+B0
0922 IF G=1 P,H80+D2+B4+B0
0923 IF G=1 P,H80+D2+B4+B0
0924 IF G=1 P,H80+D2+B4+B0
0925 IF G=1 P,H80+D2+B4+B0
0926 IF G=1 P,H80+D2+B4+B0
0927 IF G=1 P,H80+D2+B4+B0
0928 IF G=1 P,H80+D2+B4+B0
0929 IF G=1 P,H80+D2+B4+B0
0930 IF G=1 P,H80+D2+B4+B0
0931 IF G=1 P,H80+D2+B4+B0
0932 IF G=1 P,H80+D2+B4+B0
0933 IF G=1 P,H80+D2+B4+B0
0934 IF G=1 P,H80+D2+B4+B0
0935 IF G=1 P,H80+D2+B4+B0
0936 IF G=1 P,H80+D2+B4+B0
0937 IF G=1 P,H80+D2+B4+B0
0938 IF G=1 P,H80+D2+B4+B0
0939 IF G=1 P,H80+D2+B4+B0
0940 IF G=1 P,H80+D2+B4+B0
0941 IF G=1 P,H80+D2+B4+B0
0942 IF G=1 P,H80+D2+B4+B0
0943 IF G=1 P,H80+D2+B4+B0
0944 IF G=1 P,H80+D2+B4+B0
0945 IF G=1 P,H80+D2+B4+B0
0946 IF G=1 P,H80+D2+B4+B0
0947 IF G=1 P,H80+D2+B4+B0
0948 IF G=1 P,H80+D2+B4+B0
0949 IF G=1 P,H80+D2+B4+B0
0950 IF G=1 P,H80+D2+B4+B0
0951 IF G=1 P,H80+D2+B4+B0
0952 IF G=1 P,H80+D2+B4+B0
0953 IF G=1 P,H80+D2+B4+B0
0954 IF G=1 P,H80+D2+B4+B0
0955 IF G=1 P,H80+D2+B4+B0
0956 IF G=1 P,H80+D2+B4+B0
0957 IF G=1 P,H80+D2+B4+B0
0958 IF G=1 P,H80+D2+B4+B0
0959 IF G=1 P,H80+D2+B4+B0
0960 IF G=1 P,H80+D2+B4+B0
0961 IF G=1 P,H80+D2+B4+B0
0962 IF G=1 P,H80+D2+B4+B0
0963 IF G=1 P,H80+D2+B4+B0
0964 IF G=1 P,H80+D2+B4+B0
0965 IF G=1 P,H80+D2+B4+B0
0966 IF G=1 P,H80+D2+B4+B0
0967 IF G=1 P,H80+D2+B4+B0
0968 IF G=1 P,H80+D2+B4+B0
0969 IF G=1 P,H80+D2+B4+B0
0970 IF G=1 P,H80+D2+B4+B0
0971 IF G=1 P,H80+D2+B4+B0
0972 IF G=1 P,H80+D2+B4+B0
0973 IF G=1 P,H80+D2+B4+B0
0974 IF G=1 P,H80+D2+B4+B0
0975 IF G=1 P,H80+D2+B4+B0
0976 IF G=1 P,H80+D2+B4+B0
0977 IF G=1 P,H80+D2+B4+B0
0978 IF G=1 P,H80+D2+B4+B0
0979 IF G=1 P,H80+D2+B4+B0
0980 IF G=1 P,H80+D2+B4+B0
0981 IF G=1 P,H80+D2+B4+B0
0982 IF G=1 P,H80+D2+B4+B0
0983 IF G=1 P,H80+D2+B4+B0
0984 IF G=1 P,H80+D2+B4+B0
0985 IF G=1 P,H80+D2+B4+B0
0986 IF G=1 P,H80+D2+B4+B0
0987 IF G=1 P,H80+D2+B4+B0
0988 IF G=1 P,H80+D2+B4+B0
0989 IF G=1 P,H80+D2+B4+B0
0990 IF G=1 P,H80+D2+B4+B0
0991 IF G=1 P,H80+D2+B4+B0
0992 IF G=1 P,H80+D2+B4+B0
0993 IF G=1 P,H80+D2+B4+B0
0994 IF G=1 P,H80+D2+B4+B0
0995 IF G=1 P,H80+D2+B4+B0
0996 IF G=1 P,H80+D2+B4+B0
0997 IF G=1 P,H80+D2+B4+B0
0998 IF G=1 P,H80+D2+B4+B0
0999 IF G=1 P,H80+D2+B4+B0
1000 IF G=1 P,H80+D2+B4+B0

```

## ③画面のハードコピー例

1978 JULY

SUN MON TUE WED THU FRI SAT

|    |    |    |    |    |    |    |   |
|----|----|----|----|----|----|----|---|
|    |    |    |    |    |    |    | 1 |
| 2  | 3  | 4  | 5  | 6  | 7  | 8  |   |
| 9  | 10 | 11 | 12 | 13 | 14 | 15 |   |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 |   |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 |   |
| 30 | 31 |    |    |    |    |    |   |

## マイコンショップ

## オーイン ヤングエキスパート募集!!

放熱器のトップメーカー水谷電機工業が、このたび内外一流メーカー及び商社の支援により、マイコンの総合ショップを、秋葉原駅前オープンしました。マイコンおよび、キット、パーツ、周辺機器にいたるまで揃え、書籍関係では、入門書から専門書まで陳列した本格的な専門総合ショップの誕生です。

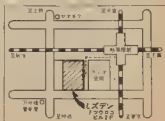
マイコンユーザーに親切でいねいな応待のできる方及びマイコンにキョウミある方を求めています。

履歴書送付のこと、面談日通知します。

身元あり、社会保険完備。待遇当社規定によります。

**水谷電機工業株式会社** **マイコン事業所**

東京都千代田区外神田1-15-6 TEL (255)4301 秋葉原駅前 ミッドビル2F



V/F

F/V

## コンバータ



ハヤマ技術部

マイコンを使って、機械の制御、ミュージック・シンセサイザなどの応用にV/F、F/Vコンバータを考えておられる方もいると思います。1/O'78年2月号にもRC4151を使ったコンバータの紹介をしました。今回は、V/F、F/Vコンバータの変換特性、応答特性を中心に実験例で紹介していきます。

## はじめに

V/F・F/Vコンバータは、共にRC4151と741を使用していて、セッティング・タイムの早い、また直線性のよいものになっています。

RC4151は、図1のようにコンパレータ、ワンショット、正帰にスイッチ

図1. RC4151ブロック図

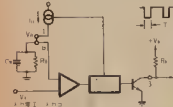
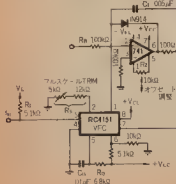


図2. 高精度V/Fコンバータ回路



される電流源から構成されています。コンパレータは、7ピンに加えられた正の入力電圧と6ピンの電圧とを比較します。入力電圧の方が高ければ、入力のコンパレータはワンショットを動作させ、その出力はロジック出力とスイッチされる電流源に加えられます。ロジック出力は低レベルとなり、同時に電流源から電流が流出します。この電流で  $R_1$ - $C_1$  回路に電荷が供給され、 $V_{in} > V_1$  となるまで続きます。これで定常状態となり、この状態を保てるように充分速い速度で  $C_1$  に電荷を供給します。

このような動作をするので、安定した発振周波数が得られるわけです。

## 実験

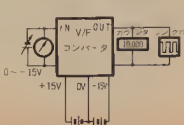
## 1. V/Fコンバータの実験

HE301の完成品を写真1に、今回路

写真1 V/Fコンバータ完成品



図3. V/Fコンバータ測定回路



図を図2に示します。電源は、 $\pm 15V$ と入力電圧用の0〜15V連続可変電源が必要です。そして測定器は、周波数カウンタ、電圧計、シンクロスコープを用意します。図3に接続図を示します。調整法は、入力電圧を0Vにして、出力が0Hzになるように $VR_1$ を調整します。次に、入力電圧を正確に10Vして $VR_2$ を調整して10kHzになるようにします。

さて、測定してみましょう。入力電圧を正確に $-9V_0 \dots -1V, \dots -0.01V$ というふうに合わせて、そのときの出力周波数を観測します。写真2、3の結果を図4に示します。上の方は、12V位まで直線性がよいようです。下の方は、0.01Vまでは10Hzとして観測できます。

## 2. F/Vコンバータの実験

HE302の完成品を写真4、全回路を図5に示します。電源そのほかは前記と同じですが、入力にパルス・ジェネレータ(出力5V<sub>p-p</sub>)をつなぎ、出力に電圧計とシンクロスコープを接続します。図6に示ように結線します。調整は、入力に何も周波数を入れない状態で $VR_1$ を調整して、出力を0Vにします。次に、10kHzを入れて出力が10Vになるように $VR_2$ を調整します。

測定に入りましょう。直線性の測定は、入力周波数を10kHz、 $\dots$  1kHz、 $\dots$

図4 入力電圧・出力周波数特性

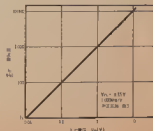


写真2. 10Vの時に10kHz



写真3. 5Vの時に5kHz

20Hzというように正確に合わせて、その場合の出力電圧を計測します。その様子を写真5、6に、結果を図7に示します。

次は、応答速度の測定です。ブロック図(図8)に示すように、スイッチを入れ出力電圧をシンクスコープで観測します。入力に10kHzを入れておきます。その結果を写真7、8に示します。セッティング・タイムが、約15msなのがよくわかります。

### ⑧. V/F, F/Vコンバータの再現性の実験

V/F, F/Vコンバータの実験を行ないましたが、その結果、相方ともかなり直線性がよいので、両方を接続して再現性があるかということについて、実験を行ないました。図9に示すように、コンバータ。各測定器を接続します。入力に一定電圧を入れ、それを周波数に変換し、さらに、その周波数を電圧に変換して、電圧の計測をします。もちろん、周波数も計測します。その

写真4. F/Vコンバータ完成品



測定を写真9,10,11に示します。そして、結果を図10に示します。次は、立ち上がり時間の測定です。入力電圧を0Vから10Vに、順時に変化させ、そのときの出力電圧の時間を測定します。写真12に示します。

## 結 果

### 1. V/Fコンバータ

入出力直線性を測定した結果、図4によると、ほぼ0.1%以内に納まって

図5 F/Vコンバータ回路図

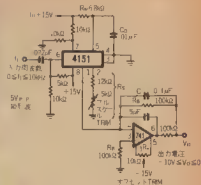
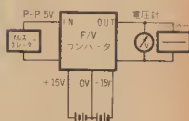


図6 F/Vコンバータ測定回路



います。RC4151のメーカーが出しているデータは、0.05%ということになっているようです。ただし、RC4151だけでは1%程度です。0V~12V位までの入力に使用できることがわかります。応答速度は、10ms位です。RC4151だけです。メーカーの資料から、135msほどです。

図7. 入力周波数/出力電圧特性

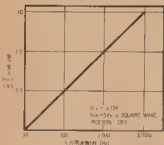


図8 F/Vコンバータ応答速度測定回路

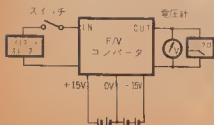


図9 V/F, F/Vコンバータ直列接続測定回路

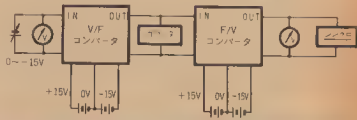


写真5. 10kHzで出力は10V

写真6. 5kHzで出力は5V



写真7. F-Vコンバータ応答速度



写真8. F-Vコンバータ応答速度



写真9. 入力に1V入れると出力に1Vで



写真10. 入力に5V入れると出力に5Vで 写真11. 入力に10V入れると出力に10Vで 写真12. 入力に10Vを入れた時の出力特性

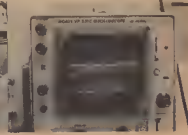


図10. 入出力電圧特性

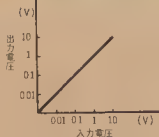
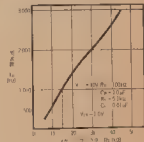


図11. スケールファクタ/温度特性例



## 2. F/Vコンバータ

入出力直線性を測定した結果、ほぼ0.1%以内に納まっています。この例でも、RC4151だけのものより、741を使用したの方が、かなり精度が良くなっています。

## 3. V/F, F/Vコンバータ直列接続

写真9~11までに示すように、ロスがなければ、入力に10V加えると、出力も10Vが出てきます。直線性についても、かなり良いようです。

今回、実験を行なわなかったのですが、ICメーカーが出している特性例を示しておきます。図11はスケール・ファクタ/温度特性、図12は温度ドリフト特性、図13は電源電圧ドリフトの特性について示します。この特性は図14の回路です。

図13. 電源電圧ドリフト特性例

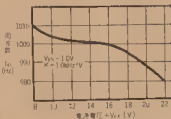
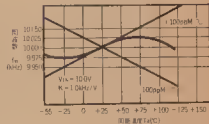


図12. 温度ドリフト特性例



## 応 用

### 1. FSK復調器

周波数-電圧コンバータの応用例として、2種類の周波数のどちらかが連続的に入ってくる場合、どちらの周波数が入ってきたかを感知し“0”、“1”の論理レベルで表示する回路です。

### 2. モーター速度制御

トランスジューサを使って、モーターの回転軸の変化をパルス信号に変えます。パルス周波数が、回転速度に比例することを利用して、F/Vコンバー

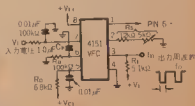
タで直流電圧を変え、モーターを制御する。

以上のような応用例もあります。

## さいごに

この実験を行なうにあたり、新日本無線の方から資料をいただき、その資料を今回使用させてもらうことの許可をいただきました。また、その他のことで、協力いただきましたので、ここで、協力していただきましたので、上にてお礼申し上げます。

図14 図11~13の測定回路





# New Products

## § シリアルプリンタ §

■NB3300バドミントンプリンタは、印字ヘッドにバドミントン羽根型ヘッドを採用したシリアル・プリンタ。従来機種NB3000の機能を高め、小型、軽量化を計ったもの。製品には、ベータ・C（電源、カバーなどを除いたOEM販売用）、受信用（RO）、キーボード付き（KSR）の3モデルがある。

### 《仕様》

▶印字速度…55字/秒▶最大印字数…136字/行または163字/行▶文字種類…128種▶印字間隔…2.54mmまたは2.12mm▶行間隔…4.23mmまたは3.18mm▶印字方式…バドミントン羽根型ヘッドによる静止活字インパクト方式▶複写能力…オリジナルを含め6枚▶寸法…幅600×奥行535×高さ210mm（KSR）▶消費電力…250VA（KSR）▶オプション…電源（内蔵可）、フロントインサータ、スプロケットブラテン、ボトムフィード機構、グラフ印字機能。

NB3300バドミントンプリンタ（RO）

NB3300バドミントンプリンタ（KSR）



《価格》 エンドユーザー価格 ￥1,030,000より

《問い合わせ先》 日本電気(株) 端末装置事業部販売促進部

〒108 東京都港区芝5-33-1 日本電気本社ビル

☎(03)454-1111

## § ハンド・ヘルド・ターミナル §

■ハンド・ヘルド・ターミナルは米国ターミナル・システム社の製品で、手持ち型の簡易CRTディスプレイ、128文字種のASCIIコードをトランスミットできる簡易キーパッドを備え、コンピュータと20mAクレント・ループ、またはRS-232Cインターフェイスを介して接続する。

機種はHT/2、HT/3、HT/4、HT/5、HT/8の5モデルがあり、表示可能文字数とバッファ・メモリ内蔵の有無により、機種が別される。

### 《特徴》

●HT/2：1000文字バッファ・メモリ付き、2行×10文字表示部、転送速度110・150・300・1200ボー切り替可、半2重/全2重切り替可、奇数/偶数/マーク/スペースの各バリエーション切り替可。

●HT/3：バッファ・メモリなし、1行×12文字表示部

●HT/4：バッファ・メモリなし、2行×12文字表示部

●HT/5：バッファ・メモリなし、状態表示用の6個×2



### 桁の表示ランプ

●HT/8：1000文字バッファ・メモリ付き、4桁×20文字表示部、その他HT/2と同様の機能あり、いずれのモデルも128文字ASCIIコードを送れるキーパッド付き

### 《価格》

HT/5 タイプ約¥22万から最高HT/8 タイプ約¥149万まで各様。写真はHT/2 タイプで¥68.8万。

《問い合わせ先》 インターニックス(株) ☎(03)369 1101

〒160 東京都新宿区西新宿7-4-7 第二太田ビル

## § 4K スタティック・RAM §

■i2141はH-MOSプロセスを用いた4K×1ビットのスタティックRAM。速度の異なる4製品と、3種類のローパワータイプを含む計7種類の製品がある。アクセス・タイムは120nsから250nsまであり、完全スタティック構造のため、サイクル・タイムも同一。全製品とも、+5V単一電源動作で、許容電源電圧変動は±10%、入出力ともすべて、TTLコンパチブル。

### 《特徴》

完全スタティックと超低消費電力のため、ローパワータイプの最大消費電力は、動作時で40mA（200mW）、スタンバイ時にはそれからさらに5mA（25mW）まで下がる。このパワーダウンには、クロックを必要とせず、単にチップ・セレクト端子のD.C.的制御だけで行なえる。

同様の高速4KスタティックRAM、2147と完全に互換性があるため、今回の2141の登場によって、ユーザは各製



品に応じた適切な製品選択が可能になった。

### 《価格》

|          | 2141-2  | 2141-3  | 2146-4  | 2147-5  | 2147-6  | 2147-7  | 2147-8  |
|----------|---------|---------|---------|---------|---------|---------|---------|
| 最大アクセス時間 | 120     | 150     | 200     | 250     | 150     | 150     | 150     |
| 最大消費電力   | 40      | 20      | 50      | 55      | 40      | 40      | 40      |
| 動作電圧     | 5V      | 5V      | 5V      | 5V      | 5V      | 5V      | 5V      |
| データ保持時間  | 100     | 100     | 100     | 100     | 100     | 100     | 100     |
| データ保持電圧  | 0V      | 0V      | 0V      | 0V      | 0V      | 0V      | 0V      |
| データ保持電流  | 0A      | 0A      | 0A      | 0A      | 0A      | 0A      | 0A      |
| データ保持温度  | 0°C     | 0°C     | 0°C     | 0°C     | 0°C     | 0°C     | 0°C     |
| データ保持湿度  | 0%      | 0%      | 0%      | 0%      | 0%      | 0%      | 0%      |
| データ保持寿命  | 1000000 | 1000000 | 1000000 | 1000000 | 1000000 | 1000000 | 1000000 |

《問い合わせ先》 インテル・ジャパン(株)

〒154 東京都世田谷区新町1-23-9

フラワービル 町町東館 ☎(03)426-9261

# マッ プ 中京地 図

## ■共同印刷 (052)951-7661

ジャーナル・プリントのタイプで、英数字48行が40行打てる。KDP-401 ¥15万。

## ■晴電子 (052)901-9293

開店して間もない。そのためか、一般のジャンク屋と比べれば整理されてる。アドレスは、41号線から少々入ったところ。バス停は池之端下車。

パーツ類は、特にC-MOS (家庭)、TTL、TV GAME LSI、CLOCK LSI、マイコン関係LSIがある。マイコンI/O用コネクタにパツダンのUSAコネクタが安い。(米国はシカゴ) アンペックス社。

|               |     |        |
|---------------|-----|--------|
| D D K57-30140 | 14P | ¥1,000 |
| 30240         | 24P | ¥1,200 |
| 30360         | 36P | ¥1,460 |
| 30500         | 50P | ¥1,620 |

## ■東名電子産業 (0582)65-2627

サリボルがなんと¥40,000まで 800Wマイコン成造器管 (ボリスレーダ見つけたら速に通信してみよう?)、ジャンクのオシロスコープ¥15,000。都市ガスのメーターまである。昔が持っているジャンク品をまとめて持って行けるが買取ってくれる。損のないように交渉しよう。いま、C-MOSとリニアICが特価中。

## ■本歩通商 (052)263-1620

ラジオ・センター2F。とうとう、ここまで来たBASICの彼。2K BASICで何が出来るか考えてみよう。いや考えてみた。アセンブラならまだしも、これは、マシン語とはもう別の世界への第1歩/第2歩と書けないわけは、歩数3万2千しか使用できないため。2K BASICを使用して何が仕事に使えるのかというキミ。この現状ではやめにして。

## ■一宮電子 (052)503-2001

5月に名古屋ラジオセンターがOPENするの、1ヵ月ほど休業中と言うことでしたが、再開しました。

## ■F Kシステム (0586)76-6701

紙テープ、6.7.8単位。プリンタ用インクリガン記録紙、紙カードなどコンピュータ周辺の紙類はすべてそろっている。事務の女性、かなりの美人) リコーコンピュータの中にもあるかも知れませんが、リコータイプボールはIBMとコンパチですらから。

## ■カマデン (052)263-0446

トランジスタ・カーブトレーサー・アダプタ ¥8,500。トランジスタだけでなく 絶縁ゲート、FET、その他の特性をすぐ見ることが出来る ジャンク品に最適。

## ■カト無線 (052)241-7336

パーツ類および一般家電品については地上6F地下1Fの当ビルカトームセン電気館建設中のため4月28日までは休業していましたが、再開しました。

## ■名古屋マイコンセンター (052)232-1851

マイコンを初歩から実用までのマイコンスクール。初心者向けマイコン・コースの13か



プログラム、インターフェイス、デジタル、LSIの5コースがある。1K ROM 2708の書き込みが手動式。紙テープとも¥600でOK。ほかのROMの書き込みもできる。

## ■サンエレクトロニクスデザインセンター (05875)4-7111

SUNPEC8000シリーズのサンプルがある。サンペックのプリント・パターンもある。プリント・パターン、技術的なことについても相談のついで。

## ■アツク電子機 (052)932-1720

タイプリタ、放電プリンタとキーボードが一体となったもの、¥99,800 ローンもOKのようです。

## ■新機電業 (052)915-1247

電子回路→プリント・パターン→製作まですべてOK。ベーク、ガラス基板によって価格が違ふ。特にプリント・パターンをあまり考えて書きたくない人にはとても良い。

## ■セイコー電機

ラジオセンター2Fジャンク店。出物のトランスやT.T.ダイオードなど良い物がある? トランスは、セルモーターも回る12V 30Aから、12V 0.5Aなどさまざま。60W×2のステレオPAが ¥4,000から。パワートランジスタは、450V 2A→¥200。ゴテルと少し数量買って、それから大社長O円です……がヒケツ。

## ■ヒケツショップ (052)263-1627

Let's Bを中心に店内デモ。キミのキャンセル・カードを見て買おう。

## ■タケイ無線

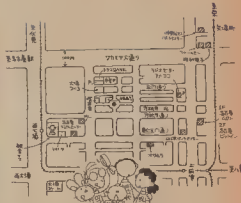
ラジオセンター1F、早くもPET2001が店頭にある。文字の安定性、値段もダンパツ。店頭では、ハゲ頭のおっさんが胸を組み考え込んでいたその横で、小学生がスラスラとプログラムを打っていた。

## ■コスモス (052)264-0005

8bit用紙テープ¥500/1個他中古端本装置多数。

## ■東海コピーセンター (0586)45-4793-7236

コピー専門店で奇天烈。第2号店(コトラストを強くしたもので、基板作成にOK)



はかに図面製本など入り口にコピーのサンブルがあり、図面との差もわかる。場所は一宮広聖体育館南側。

## ■ビットイン名古屋 (052)263-0971

以前、高校生が割と来ていたようですが、先日、日曜日の昼とあってBASICのできるマイコンはかなり人気がありました。中には、小学生も少しですが来ていました。やはり、BASICの強みでしょう。マシン語のように専門知識が必要でなく、要領がわかりやすく、ムダなくプログラムができるからでしょう。また、ビットインでは、マイコンクラブの会費を募集しています。

P-ROMの書き込みは無料。PD454をキーボードを通して打った場合、256バイト平均1時間と言うことでした。

## ■東海フラックス研究所 (052)622-5255

通販を主にしていますが、店頭販売も行なっている。ただし、これはユニットの取寄で基板のみは行っていない。休日は日曜日。TV 2chの水晶発振器が¥10,000電源トランス¥3,000。

## ■大越電機工業製作所 (04803)8-0065

プリントパターンを自分で割けば安く、プリント基板を作ってくれる。ベーク基板の場合1cm<sup>2</sup> ¥3でボールペンでパターンを書いて送っても良い。ただし、線幅とその間隔は厚さ0.5mm以上のこと。それ以下の場合には製作困難で返されて来ます。ICの足の間に2本以上のパターンを走らせることはできず、1本ならOK。穴は1個 ¥1。ガラス基板は1cm<sup>2</sup> ¥6。両面基板もできる。

(正栄)



あきはばら地図

みなさん元気ですか、あきはぼらマ  
ブも回を重ねること20回

これも担当者の努力のしるもの。いや実は、I/O編集部のシゴキのたまものです。(コレはホントヨ)

さて、みなさん、このあきはばらに、  
いったい何軒のマイコンショップがある  
かご存知ですか?』種々このころは、  
これを書いている本人も知らないの  
です。駅前のあるビルの中には、マイ  
コンに関係した店が9店もあるし、そ  
れから考えたとあきはばら全体では、  
いったい何店あるのか、チョノトわ  
かりません。あー。

さて、そろそろ本題に入りましょうか。今月は部品整理学その1(注“その1”とは“その2”があるということです。)と題してお話しましょう。

みなさんのお宅では、どのように部  
品を管理しておられますか。

つい先日まで、私のところでは、大  
学生協の特売で買ったレターケースに

入れてあったのですが、増えつつける部品のため、ついにパンクしてしまいました。そこで、ほかに何かよい整理術をさがしたところ、あきはばら通りの工具屋さんで売っている部品ケースがなかなか便利で都合よく出来ているんですよ。

ものは、宝山で作っているB-40FGというケースです。サフィノタスのFとかGというのは中箱の仕切の大きさを示すもので、Aから始まり、Gまであります。Gは2段重ねになっています。詳しくは図を見てください。

そうそう、このケースを安く売っているところは、牛久屋さんの通り、加越電機との交差点にあるヒロセパーツセンターの2階で、他の店よりも100円安い。

(ついでに、ネンを整理するのに都合のよいケースが150冊で決っています。)

さて、こうして丁に入れたケースに部品をどう整理するか。私の場合、45

ルス回路しかいしらないので、トラン  
ジスタはPNPとNPN、大電力か、小  
電力かなどということを中心にして区  
別しました。(型番は外にします。H)

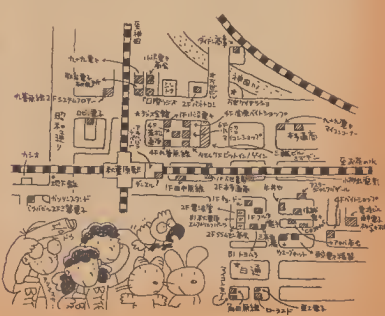
コンデンサは、平滑用、バイパス用といった具合にわけます。

抵抗は、ひとつの抵抗値にひとつのケースを割り当てたら大変なので、近い値のものを集めて整理します。

ダイオードもトランジスタと同様に用途によってわけます。

デジタルICは型番によって区別します。TTLなどは発泡スチロールの板に、MOSは半導体マントの上にのりつけておきます。

チョットまで、ここまで読んでくれた方は気づいたと思うのですが、この作者は、部品の整理等などとえらそうな題をかかけて、その実、内容は自分のところで整理しているやり方をのべている。イーやまいった様心をつかれました。ア、ハ、ハ、ハ、とぶてて笑





- 主・横 296 奥行 32 高さ 33mm×3列
- 仕 切: 23mm 間隔
- 仕切筋: 9本つき

| 品 番    | 内箱の組合せ     |
|--------|------------|
| B-40EE | E形 2個      |
| B-40EF | E形・F形各1個   |
| B-40EG | E形1個, G形2個 |
| B-40FF | F形2個       |
| B-40FG | F形1個, G形2個 |
| B-40GG | G形4個       |

ケース外寸: 290×220×45 mm

てゴマかす。)

しかし、みなさん部品の整理など自分が必要なときに、すぐにその部品が取り出すことができる必要かつ充分なのです。論理に強いI/O読者のみなさんならおわかりでしょう。

製作記事やデータなどの整理についても同じことが言えます。

(蛇足ですが、私のところでは、マイ

コン、電源などの項目と各項目に共通するものという具合に分けています。]

## 余 談

このあきはばらマップの担当者は、低抗やコンデンサなどがいっぱい入っているジャンク袋が好きで、よく買います。先日、I/O ラボ（コンピュータラビIIのよこ）に遊びに行ったときのことです。I/O ラボの主人池上さん曰く、「〇×△さん、R電子で売っているコンデンサのつまった袋はイイよ。M社のコンデンサは、つぎの日の朝でも放電するヨ。」といってビックリしました。これには私もビックリしました。

しかし、話ほこれで終わったわけではないのです。その帰り、コンピュータラビIIのカギをガチャガチャいじくりまわしているあやしい男を見つけました。おそろおそろ、何をしているかなおねたら、こはトイレではないのかとのたまう。小生、あけらかんとして曰く、「こはコンピュータショップですヨ?」

【後で、関係者に聞いたのだが、時々、

まちがわれるそう。なかには、目の前で、チャックをおろし始めた紳士もいるそうだが……。]

しかし、いくらジャンボが近くにあるといってもトイレとマイコンショップを間違えうとは……。]

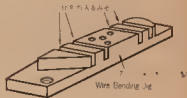
## "Wire Bending Jig"

先B、Radio Shackでおもしろいものを見つけました。

それは、Wire Bending Jigと呼ばれる針金折りまげ器です。

図に示すような形になっていて、かなり太い針金もまげることができま。お値段は 250円。

私はこれで、チエの輪をつくって遊んでいますHi、 (N)



●ロビン電子では、9080Aが¥2,600。(とうとうCPUは3千円を切ってしまった…) MC 6802が¥ 8,550で売られている。4 KスタティックRAM TMM 314P (2114タイプ) は¥2,000. 2102ローバワーが¥ 450となっていた。不良品があったら取り替えてくれるとのことでした。

●さて、話がかわるけれど、だれもが知っている1 KスタティックRAM 2102のローバワ品種の見分け方をそっとI/O読者にお教えらうのだ。ローバワタイプには21L02、2102L、2102AL (NEC製) があるけれど、ほんとうのローバワ品は前の2つ。これは電流が20~30mAくらい流れる。最後のALは、70mAくらい流れるからセミ・ローバワといったところ。

●ラジオデパート1Fの第 カマタはTDKフェライト・コアが豊富に置いてある。コアの材質は、L、H5A、H5Bでサイズ別の値段は、

| サイズ | L材   | H5A    | H5B    |
|-----|------|--------|--------|
| 20  | ¥250 | ¥450   | ¥650   |
| 30  | ¥300 | ¥600   | ¥800   |
| 40  | ¥450 | ¥890   | ¥1,090 |
| 50  | ¥600 | ¥1,220 | ¥1,420 |
| 60  | ¥680 | ¥1,400 | ¥1,600 |

サイズとは、コアの長さで単位はmm。もちろコアはE1型、トロイダル型もあった。インバータ・タイプのスイッチング・レギュレータを作ろうノ一んちゃで、格調高くせる人は一度のぞいてみたら。

ところで、I/O C1Aとは何んのことだ… (M) [I/O C1Aとは……数万人のI/Oの読者の皆様に「情報部員」となってもらい、全国各地のパーツ屋さんの「アナ場情報を集める組織(?)」です。ぜひ、「同志」として協力してねー編ー]

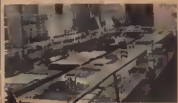


(江東区・横戸麻沙人)

## NEW SHOP

マイコンショップ「ミズデン」が、5月8日にオープンしました。場所は、秋葉原ラジオ会館となりのミツウロコ・ビル2Fです。各メーカーのマイコンをはじめ各種周辺機器をそろえています。デモンストレーション用にマイコンゲームをいくつか置いてあるので、一度立ち寄ってみては?

☎(03)255-4301





## なげなばし地図

君は、もう、BASICと遭遇したか？  
マイコンショップでは、BASICだらけといった感じで、どこへ行ってもゲームをやっている。

ところで、かく言う私めも、マイコン・ボードなるものを、ついに買ってしまったのじゃ。SWバチバチから開放されたものの、V-RAMを自作したために、えらくしんどかった。でも、今はTVでゲームをやって遊ぶほど、余裕ができてきた。まだ4KBASICは走らないけど、マイクロBASICなら、一応のことはできるのだ。

グワシ！

それから、雑誌屋さん達にお願ひ、あらぬ誤解を招くようなものはつづいてくださいますませ 類似品に注意！

著者が10年来、うろつき歩いて書いたこの地図の類似品が、出回っているようです。それで、というわけではないのですが、「元祖にっぽんばし地図」としまして、今月はこの地図にのっていない店もとりあげることになりました。

### ■本州商会

日本電販の、一つ雨の筋を東に入った所に山田ビルがあります。その4階にあるのです。(入口の奥に、エレベーターがあります)行ってみるとレタイクがたくさんならべてありました。

この本州商会は、中古品を扱ういわゆるジャンク屋さんですが、ミニコンからブルトーザまでやっているそうです。Q. ちょっと、ここはわかりにくいですがね。

A. ええ、でも会社関係を主にやっているの、個人のお客さんはどうも……

一度、マイコン関係の雑誌に広告を出したんですが、変な問い合わせばかりかってくるので、やめてしまいました。

Q. マイコンの周辺に使えるようなものはありますか。

A. そうですね、このASR-32はどうですか？BAUDOTコードですが

安いですよ。ASCIIコードのASR 33もありますが、割高になりますね。それに、表面がプラスチックなので割いですね。こちらのオリベッティのTTYVなんかは、がんばりようでいいですよ。

それから、ファックスなんかもマイコンにつないだらおもしろいと思いますね。(ウウウ、著者がやろうと思っていた事を先に言われてしまった)

Q. ファックスの安いのありますか。

A. いろいろありますが、まあ安いのは松下で、4万ですね。B6の大きさです。ここにはないんですけど、大阪電研サービスにある7万のと同じです。品物が多いので、電研サービスに聞かせてもらっているんです。見てみますか。

と、言いながら電研サービスに案内してくれました。ちょうど常連らしいお客さんが来ていたので、留守番をしてもらいました。

本州商会 (手前がASR-32、奥がオリベッティ)





P-R OM  
1702値段は1702+500円(??)

#### ■大阪バイトショップ

4027.4とコンパチの東芝の、4KマイニットRAMが入荷しました。  
¥1,000です。

東芝からTMP-9080Aを使ったマイコン・ボードが発売されました。TLCS-80Aの、EX-80というボードです。今までのこの種のボードと大きく違っているのは、ボード上にTVディスプレイ回路が組み込まれていることです。キャラクタにも、グラフィックにも使えます。同期回路には専用のLS1, TC5003が使われています。それにLED表示器と、16進のキーボードもついているのです。考えるにキャラクタを出すのに16進のキーボードを使うのかな、そうするとキー数が少ないからめんどいことになりそうだが……と思って、パンフレットの写真をみると、ビット・パターンしか出して



(大阪市・馬場隆征)

いないのです。やはりLEDも付いているので、モニタはTVベースのものではないと考えられます。しかし、これだけ付いていても万そこそこだということなので安いような気がします。

#### ■同本無線

こはTK-80BSと、H68/TRのBASICのデモをやっています。H68のBASICは4Kで、テープでディスプレイボードに付てきます。ユニークなところもあるのですが、\*、\*、\*、\*、\*、\*の使い方に気を付けなければなりません。実行速度が少し遅いように思えました。やはりこれは、あくまで一時しのぎで、12K-ROMのBASICが本名でしょう。

ここかなり常連の人が多くて、いろんな人が入れかり立ちかわりやっています。

しかし著者がBASICのプログラムを作ってキーを押していると、横でシンセサイザをいじっていたりして

いるのです。まあ、ピンク・フロイドとまではいかなくても、少なくとも雑音を出してくれば、著者もずっと気持ちよくプログラムを作れたのですが、これがまたムチャクチャなので頭に来ます。どうかシンセサイザをやろうとする人、音楽の勉強も怠りなさいよう。

3端子電源も、ついに10Aが出た  
HA78P05C 5V ¥4,300  
MB 8111N ¥740

#### ■上新電機

LSの200番台の1Cがあったのです。これは筆者の大見落としです。筆者がV-RAMを作る時に、上料にはないだろうと思って探し歩いてあきらめたのに……新にあったなんて……おもしろいしました。

LS240, LS241, LS244 各¥805  
LS273 ¥610, LS168 ¥755  
LS165, LS166 各¥835  
シナティック SY2114 ¥2,500

同本無線のみなさん+α



アドテックのTV-D-03用の、細面競争ゲーム(トンパースト) ¥3,500  
16K RAMボードもあるよ。

PETもおいてあるよ

◇ブラウン管 150 CB-1

新品¥1,000 (ナニワ商会)

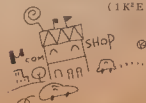
・テクニカル・サンヨーに、三菱の、4KRAMがあった。セラミックのメタルシルでピカピカ、とてもきれいなのです。

MS6724S ¥2,500

◇東海電機は、最近LEDが、よく店頭に並びます。7セグメント・2桁のGL-7P-201 アノードコモンが、¥480

◇大阪抵抗社の北隣の電波堂には、ラジオの他に、TVゲームがたくさんある。最新のカートリッジ式のもの(アタリ社)もありました。

(1K²E1)



#### ■共立電子産業

TK-80BS, SWTPC-6800, SUNPEC-8000TKに今度TRS-80と、PETを加えて、計5台のマシンが同時に動いているのです。これだけ並ぶと、壮观です。スター・トレックや、ボールが、ピコピコ動くゲーム(PET)など、小さな子供もいて、きながらゲーム・センターといった趣がありますが、なかなか精力的にガンバっています。

TMS1121を使った、タイマープロセッサもありました。4つの装置を、1分-1週間、20のプログラムが可能で、これを時計の方に置くべきか、プロセッサの所に置くべきかといったが、カウンタの上に落ちついたようです。

LM567 PLL 1C ¥800

\*

KEY-75(JISコード 75キー)

¥9,200

JISのエンコード(364D-01)が、おいてある。

ASCIIをJISに変換できる

## ■次号予告

6月25日発売の次号では、CRTディスプレイの紹介の他、ワンチップCPU、マイコンの制御への応用などを掲載する予定です。なお、ミスターXも今回は登場しますのでファンの方はご安心を!

## ■編集後記

▶読者の皆様、いつも多くのお便りありがとうございます。最近のマイコン界は、いわゆるワンボードもののブームはようやく落ち着き、マイコン本位の使われ方つまり、①制御用に組み込まれ、従来のTTLなどとの置き換え、または高級化をはかる。②ホーム・コンピュータとして、実用化する、という2つの方向に分極化してきたようです。

いままでのマイコン・ファンとしては、ただだコンピュータをつくるのに全力をあげる傾向がありましたが、今後は、より幅広い知識が要求されるようになると思います。つまり、「どのようなマイコンをつくるか」という時代から、「マイコンをどのようにつかうか」という時代に移りつつあるのです。I/Oも同様なことが、これらの要求にこたえるべく努力していきます。

▶そして、読者の皆様にお願ひですが、「自分はマイコンをこのように使っている」という例がありましたらぜひ投稿してください。きっと他の読者の参考になると思います。たとえば、模型を動かすのにマイコンを使ったとすると、技術的にはマイコンによるリレーあるいはサリスタの制御方法であったりするでしょう。その技術は、きっと他の応用を考えている方にも役立つと思うのです。皆様の投稿をお待ちしています。

## ■お知らせ

I/Oは記事に具体性を増すため、今回路線、全リスト公開などをしつつやるため、このところ約40ページ増ページして、特価に定めてしまっています。括弧を小さくして載せるのも一つの手なのですが、印刷屋さんから文句を言われる限界まで来てしまいました。といって、今回路線、全リスト掲載の方は止めるつもりはまったくありませんので、今後ともこの方針をつらぬくため、申し訳ありませんが、定価を380円とさせていただきます。読者の皆様のご理解とご支援をお願いいたします。

なお、定期購読の方は現行料金に据え置きです。

## ■原稿募集

「I/O」はみんなの広場です。以下の各原稿を募集していますので、ぜひあなたも参加して下さい。

- ①イベント、ミーティング、講習会、勉強会 etc のお知らせ。
- ②製作・実験のレポート 原稿用紙(400字詰 横書き) 5枚くらいにまとめる。図、表はエンピツ書きでO.K. 写真もぜひ入れて下さい。
- ③「I/Oポート」のマイコン・クラブ紹介(メンバーの写真も!)
- ④秋葉原の情報(お買物品の情報 etc.)
- ⑤ソフトウェア道場 プログラムの説明とアセンブラまたはマシン語のリスト、フローチャートも。

②~⑤は採用の場合には稿料をさしあげます。

なお、投稿の際には以下のことを必ず記入して下さい。

- (イ)現在の所属(ペンネームの場合でも一応ご記入願います。)
- (ロ)連絡先(勤務先または自宅)の住所、電話番号。
- (ハ)年齢、学年
- (ニ)現在所有しているマイコンがあればその名称(例: 8080, 6800, SC/MP)

編集部に対するご意見がありましたら、あわせて、お寄せ下さい。

## ■投稿先

〒151東京都渋谷区代々木2-5-1 羽田ビル507 工学社内  
日本マイクロコンピュータ連盟「投稿係」

## ■定期購読のおすすめ

予約申し込みは 半年、1年で、半年以上申し込まれた方、**「マイコン連盟」**の会員として登録されます。

①1冊450円(送料込)

②半年・2,300円(送料込)

③1年・4,300円(送料込)

## ■団体割引

なお、5名以上で1年間の予約をする場合は団体会員として、1名当り年間4,000円をお支払い下さい。

## ■送付方法

①郵便振替(東京2-49427)

振の通信欄に、何月号からご希望が明記してください。

②現金書留 } 何月号からご希望が明記したものを、同

③定額小為替 } 封してください。

のいずれか。

●なお、継続して申し込まれる方は、会員番号も忘れずにお書きください。

## ■送付先

〒151東京都渋谷区代々木2-5-1 羽田ビル507 工学社内  
「日本マイクロコンピュータ連盟」

▶他社の名前で雑誌の輸入をやって、しかも代金も払わない者がいる。これが、国際的に問題になっているが、このような犯罪者がマイコンの世界にもいるのは、なんともやりきれない。▶国際間の取引はそれ自体、お互いに相当の信頼感がなければ成り立つものではない。このような、モラルに欠けた人間が、たとえ少数でもいることは、日本のマイコン雑誌全体の評価にもひびくだろう。1日も早くマイコン界から一掃されることを望む。▶このせちがらい世の中ではあるが、せめて、マイコン・ホビーの世界だけは、お互いに正々堂々と誠実にやっていきたいものだ。(I/O)

I/O

発行人

編集人

編集

発行所

1978年6月号 第3巻第6号(通巻第20号) 昭和53年6月1日発行(毎月1回発行)

星 正明

森 昭助

日本マイクロコンピュータ連盟

株式会社 工学社

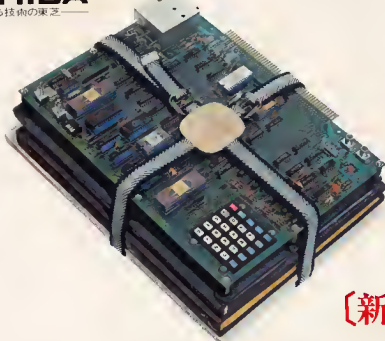
〒151 東京都渋谷区代々木2-5-1 羽田ビル507 ☎(03)375-5784 振替口座東京5-22510

印刷: 精文社

定価 380円

# TOSHIBA

——明日をつくる技術の東芝——



## 〔新発売〕

東芝マイコン・キット

## マイコン・マニアの新しい“テキスト”EX-80

東芝の新しい8ビットマイコン・キットTLCS-80A・EX-80には、テレビインタフェース回路とオーディオカセットインタフェース回路が実装されています。このため、組立後、お手持ちのテレビやラジオを利用して、即応用へ移行できる一歩進んだ形のマイコン・キットです。このEX80は、組立と応用を通してマイコンのハードとソフトが理解できるいわばマイコンの新しい“テキスト”。

外部インタフェースを加えることにより、さらに高度な応用も楽しめるので、マイコンの“入門者”だけでなく“マニアの方々”にも満足いただけるワンボード・マイコンです。第2世代のマイコン・キットEX-80で、限りなくひろがる“マイコンの世界”へあなたも旅立ってください。

### EX-80の特長

- ★CPU(TMP9080A0)を中心にした8ビットLSIファミリーと各種部品で構成された完全部品キット。
- ★テレビインタフェース回路、オーディオカセットインタフェース回路もワンボード上に実装。
- ★16進キーボードを装備。キーボード使用の際はテレビの音声回路を利用してキー入力確認ができます。
- ★組立後、即稼働できるようモニタープログラムがROM(TMM331AP)に書込まれています。
- ★カンサシシティ標準規格に準拠しているため、普通のオーディオカセットを使用してプログラム・データの書き込み・読出ができます。



テレビ  
インタフェース

オーディオカセット  
インタフェース



標準構成

標準価格 85,000円

マイコンのご相談は 東芝マイクロコンピュータ技術相談室

**マイコン セブン**

〒101 東京都千代田区外神田3-13-7 ニュー・カクタX1ビル 5F

TEL(03)255-7588~9

10:00A M~6:00P M (日・通水曜日・水曜日定休)

東芝マイクロコンピュータ・キット TLCS-80A・EX-80

# EX-80

## “Micro Computer Kit”



東芝

東京芝浦電気株式会社半導体営業推進部 〒210 川崎市幸区洲川町72 TEL(044)522-2111(大代)



## マイコンの飛躍。

より高度に、より本格的に。

マイコンの世界を、さらに広げる。  
TK-80BSとTK-80/80Eの  
コンビネーション

NEC日本電気の総合力から生れたLSI技術、コンピュータ技術の粋を要め、いま、いっぴくマイコンの世界が飛躍しました。ベーシックステーションTK-80BSとTK-80/80Eのコンビがそれ。フルキーボード付CRTディスプレイ機能を持たせたパワフルなシステムです。

会話型言語“BASIC”をはじめ、最新鋭の機能を満載したマイクロコンピュータ。専門家でない方にも、比較的容易に、目的にあわせたアプリケーションプログラムを組むことが可能です。

ホビーからビジネスまで、幅広い分野での応用、活用にチャレンジしてください。

(アマチュアホビー用に各種小規模業務用・エンジニア/プログラマーにシステムコンピュータとして・実験制御装置に・学校教育用に・パーソナルコンピュータとして)

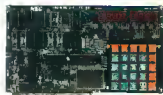
幅広い用途に広げる  
特長のかずかず

- やさしい会話型言語“BASIC”が待機時間で使用可能
- 家庭用のTVが、そのままディスプレイに
- 他に例を見ない豊富な文字とユニークな表示パターン
- グラフ表示も簡単なNEC BASICは、機械語サブルーチンとリンク可能
- プログラム開発ツールにも利用可能
- カセットテープで、プログラム、データのセーブ・ロード可能
- 5KバイトのRAMを実装済。さらに2Kバイトまで拡張可能

NECマイクロコンピュータ  
μCOM Basic Station TK-80BS

価格:128,000円 送料:1,300円(電源除く)

※TK-80/80Eと組み合わせて使用します。



完全部品キット/基本プログラムはROMに書き込み済/TTYは不要/市販のテープレコーダにプログラム録音可/プログラムデバッグ可能

エレクトロニクス  
NECマイクロコンピュータ  
μCOM Training Kit TK-80E  
価格:67,000円 送料:1,000円(電源除く)

**NEC**  
日本電気

電子デバイス販売事業部  
マイクロコンピュータ販売部  
東京都港区芝5丁目33番7号  
(徳栄ビル)〒108  
TEL (03) 454-5511(大代)

**Bit-INN**

- Bit-INN東京 〒101 東京都千代田区外神田1-15-16 ラジオ会館7F ☎(03)255-4575-6
- Bit-INN大阪 〒542 大阪市南区難波新地6番町10-1 マスザキヤビル4-5F ☎(06)647-2747-8
- Bit-INN名古屋 〒460 名古屋市中区大須4-11-5 杏林産産ビル2F ☎(052)263-0971

- 通信販売取扱店 ●東日本通信販売 日本電子販売部 〒101 東京都千代田区外神田1-16-1 万世ビル3F ☎(03)255-4571(代)
- 西日本通信販売 三ヶ谷商事 〒540 大阪市東区島町2-5 ☎(06)942-1941(代)
- 中部地区通信販売 萩原電気 〒461 名古屋市中区東桜2-3-3 ☎(052)531-3511(代)

●なお詳細な資料を希望の方は、資料請求券を随付のうえ、本書でNEC日本電気 電子デバイス販売事業部 マイクロコンピュータ販売部 までお申し込みください。

雑誌1473-6

定価

三八〇円

資料  
請求券  
I/O 8-6

